หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

เอกสารอ้างอิง

๑. กฤษณพันธุ์ สุพรรณโรจน์. คอมพิวเตอร์ในละแวกวงธุรกิจ. พิมพ์ครั้งที่ ๒. กรุงเทพมหานคร : สำนักพิมพ์แพร่วิทยา, ๒๕๒๐.

๒. ขวัญชัย คณะรัตน์. โคบอลประยุกต์. กรุงเทพมหานคร : เจริญผล, ๒๕๒๔.

๓. ขวัญชัย คณะรัตน์, สุภาพร ปาจริยานนท์, ประเวทย์ ศรีสุนทร และสิบพร ถาวรฉันท์. การวิเคราะห์และออกแบบระบบงานคอมพิวเตอร์. กรุงเทพมหานคร : ทวีกิจการพิมพ์, ๒๕๒๑.

๔. ขวัญชัย คณะรัตน์ และ คณะ. ระบบประมวลผลข้อมูลแบบสื่อสาร. กรุงเทพมหานคร : ศูนย์ประมวลผลแห่งประเทศไทย สำนักงานสถิติแห่งชาติ, [ม.ป.ป.]

๕. จุฬาลงกรณ์มหาวิทยาลัย, สถาบัน. "งานสหกรณ์กับการพัฒนาเศรษฐกิจและสังคม". กรุงเทพมหานคร : จุฬาลงกรณ์มหาวิทยาลัย, ๒๕๑๕.

๖. สหกรณ์ออมทรัพย์ข้าราชการจุฬาลงกรณ์มหาวิทยาลัย จำกัด, "คู่มือสหกรณ์ออมทรัพย์-ข้าราชการจุฬาลงกรณ์มหาวิทยาลัย จำกัด". กรุงเทพมหานคร : โรงพิมพ์จุฬา-ลงกรณ์มหาวิทยาลัย, ๒๕๒๓.

๗. ____. "ที่ระลึกครบรอบ ๒๐ ปี สหกรณ์ออมทรัพย์ข้าราชการจุฬาลงกรณ์มหาวิทยาลัย จำกัด ๑๓ มกราคม ๒๕๒๔" กรุงเทพมหานคร : บริษัทสารมวลชน จำกัด, ๒๕๒๔.

๘. Citaenbaum, Ronald L, ; and Sackman Harold. On-line Planning towards creative problem-solving. USA. : Prentice-Hall, Inc., 1972.

๙. IBM. Customer Information Control System / Virtual Storage (CICS/VS) Application Programmer Reference Manual (Command Level). New York : International Business Machines Corporation, 1978.

๑๐. IBM. DOS/VS Access Method Services User's Guide. New York : International Business Machines Corporation, 1977.

ภาคผนวก ก.

เขตข้อมูลของระเบียน

ตารางที่ ก.๑ แสดงเขตข้อมูลของระเบียนรายชื่อสมาชิก

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๖	๖	รหัสประจำตัวสมาชิกใหม่	
๒	๗	๑	รหัสแสดงสภาพจริง/ไม่จริงของระเบียน	
๓	๘	๑	รหัสแสดงสภาพถาวร/ชั่วคราวของระเบียน	
๔	๙	๑	รหัสแสดงสภาพเก่า/ใหม่ของระเบียน	
๕	๑๐	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๖	๑๑-๑๔	๔	เลขประจำตัวสมาชิกเก่า	
๗	๑๕-๗๔	๖๐	คำนำหน้าชื่อ ชื่อนามสกุลภาษาไทย	
๘	๗๕-๑๑๔	๔๐	คำนำหน้าชื่อ ชื่อนามสกุลภาษาอังกฤษ	
๙	๑๑๕	๑	รหัสเพศ	
๑๐	๑๑๖-๑๑๘	๓	รหัสแหล่งจ่ายเงินเดือน	
๑๑	๑๑๙-๑๒๕	๗	รหัสบัญชีถือจ่าย (เลข จ.๑๘)	
๑๒	๑๒๖-๑๓๒	๗	รหัสประจำตัวข้าราชการ	
๑๓	๑๓๓	๑	รหัสการเป็นสมาชิก	
๑๔	๑๓๔	๑	รหัสเงื่อนไขการเปลี่ยนคำนำหน้า ชื่อ ชื่อนามสกุล	
๑๕	๑๓๕-๑๔๒	๘	ปี เดือน วันที่อัพเดท	
๑๖	๑๔๓-๑๖๐	๑๘	(ว่าง)	

ตารางที่ ก.๒ แสดงเขตข้อมูลของระเบียนสถานภาพของสมาชิก

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๖	๖	รหัสประจำตัวสมาชิก	
๒	๗	๑	รหัสแสดงสภาพจริง/ไม่จริงของระเบียน	
๓	๘	๑	รหัสแสดงสภาพถาวร/ชั่วคราวของระเบียน	
๔	๙	๑	รหัสแสดงสภาพเก่า/ใหม่ของระเบียน	
๕	๑๐	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๖	๑๑-๑๘	๘	ปี เดือน วันที่เข้าเป็นสมาชิก	
๗	๑๙-๒๑	๓	อายุสมาชิก	เดือน
๘	๒๒	๑	รหัสแสดงสถานภาพสมรส	
๙	๒๓-๒๘	๖	รหัสประจำตัวสมาชิกคู่สมรส	
๑๐	๒๙-๓๑	๓	รหัสหน่วยงานที่สังกัด	
๑๑	๓๒-๓๔	๓	รหัสหน่วยงานที่ทำงาน	
๑๒	๓๕	๑	รหัสประเภทสมาชิก	
๑๓	๓๖-๓๗	๒	จำนวนเดือนที่เหลือที่จะกู้ใหม่ได้	
๑๔	๓๘-๔๒	๕	วงเงินขอกู้ (สามัญและฉุกเฉิน)	บาท
๑๕	๔๓-๔๕	๓	รหัสสภาพการกู้	
๑๖	๔๖-๕๐	๕	วงเงินค้ำประกัน	บาท
๑๗	๕๑-๕๕	๕	จำนวนเงินค้ำประกันที่ใช้แล้ว	บาท
๑๘	๕๖	๑	จำนวนคนที่ค้ำประกันให้	
๑๙	๕๗	๑	จำนวนระเบียนที่ใช้ค้ำประกัน	
๒๐	๕๘-๖๒	๕	เงินเดือน	บาท
๒๑	๖๓-๖๖	๔	รายได้อื่นรายเดือน	บาท
๒๒	๖๗-๗๐	๔	เงินสะสมส่งรายเดือนตามจริง	บาท
๒๓	๗๑-๗๗	๗	รายจ่ายอื่นรายเดือนที่หักโดยหน่วยงาน	บาทสตางค์
๒๔	๗๘-๘๒	๕	ยอดเงินปรับปรุงเพื่อลดรายจ่ายอื่นรายเดือน	บาทสตางค์
๒๕	๘๓	๑	รหัสเงื่อนไขการพ้นสภาพการเป็นสมาชิก	
๒๖	๘๔-๙๑	๘	ปี เดือน วันที่พ้นสภาพการเป็นสมาชิก	
๒๗	๙๒-๙๙	๘	ปี เดือน วันที่อัพเดท	
๒๘	๑๐๐-๑๒๘	๒๙	(ว่าง)	

ตารางที่ ก. ๓ แสดงเขตข้อมูลของระเบียนเงินสะสม

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๖	๖	รหัสประจำตัวสมาชิก	
๒	๗	๑	รหัสแสดงสภาพจริง/ไม่จริงของระเบียน	
๓	๘	๑	รหัสแสดงสภาพถาวร/ชั่วคราวของระเบียน	
๔	๙	๑	รหัสแสดงสภาพเก่า/ใหม่ของระเบียน	
๕	๑๐	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๖	๑๑-๑๓	๓	เงินสะสมรายเดือนตามเกณฑ์	บาท
๗	๑๔-๑๗	๔	เงินสะสมรายเดือนตามจริง	บาท
๘	๑๘-๒๑	๔	เงินสะสมรายเดือนก่อนการเปลี่ยนแปลง	บาท
๙	๒๒-๒๗	๖	ปี เดือน ที่เปลี่ยนแปลงการส่งเงินสะสม	
๑๐	๒๘	๑	รหัสการเปลี่ยนแปลงเงินสะสม	
๑๑	๒๙	๑	จำนวนครั้งการเปลี่ยนแปลงเงินสะสม	
๑๒	๓๐	๑	รหัส เงื่อนไขการคำนวณยอดเงินค่าหุ้น	
๑๓	๓๑-๓๖	๖	ยอดเงินค่าหุ้น	บาท
๑๔	๓๗-๔๓	๗	ยอดเงินเพื่อคิดเงินปันผล	บาท
๑๕	๔๔-๕๑	๘	ปี เดือน วันที่อัพเดท	
๑๖	๕๒-๖๔	๑๓	(ว่าง)	

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ก. ๔ แสดงเขตข้อมูลของระเบียนเงินกู้

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๖	๖	รหัสประจำตัวสมาชิก	
๒	๗	๑	รหัสประเภทการกู้	
๓	๘	๑	รหัสแสดงสภาพจริง/ไม่จริงของระเบียน	
๔	๙	๑	รหัสแสดงสภาพถาวร/ชั่วคราวของระเบียน	
๕	๑๐	๑	รหัสแสดงสภาพเก่า/ใหม่ของระเบียน	
๖	๑๑	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๗	๑๒-๑๘	๗	รหัสเลขที่สัญญากู้	
๘	๑๙-๒๖	๘	ปี เดือน วันที่ทำสัญญากู้	
๙	๒๗-๓๒	๖	จำนวนต้นเงินกู้	บาท
๑๐	๓๓	๑	รหัสการคิดต้นเงินชำระคืน	
๑๑	๓๔-๓๘	๕	ต้นเงินชำระคืนงวดแรก/งวดสุดท้าย	บาท
๑๒	๓๙-๔๓	๕	ต้นเงินชำระคืนงวดคงที่	บาท
๑๓	๔๔-๔๙	๖	จำนวนเงินยอดลูกหนี้คงเหลือ	บาท
๑๔	๕๐-๕๒	๓	จำนวนงวดชำระทั้งสิ้น	
๑๕	๕๓-๕๕	๓	งวดชำระที่	
๑๖	๕๖-๖๑	๖	ดอกเบี้ยชำระงวดแรก/งวดสุดท้าย	บาทสตางค์
๑๗	๖๒-๖๗	๖	ดอกเบี้ยชำระงวดปัจจุบัน	บาทสตางค์
๑๘	๖๘-๗๒	๕	ส่วนลดดอกเบี้ย	บาทสตางค์
๑๙	๗๓-๗๕	๓	อัตราดอกเบี้ยร้อยละต่อปี	
๒๐	๗๖-๘๒	๗	ดอกเบี้ยสะสม	บาทสตางค์
๒๑	๘๓-๘๘	๖	รหัสประจำตัวสมาชิกผู้ค้ำประกัน	
๒๒	๘๙	๑	ลำดับที่ของระเบียนผู้ค้ำประกัน	
๒๓	๙๐-๙๘	๙	รหัสโฉนดที่ดินที่ใช้ค้ำประกัน	
๒๔	๙๙	๑	ลำดับที่ของระเบียนหลักทรัพย์ค้ำประกัน	
๒๕	๑๐๐	๑	จำนวนโฉนดที่ใช้ค้ำประกัน	
๒๖	๑๐๑-๑๐๖	๖	มูลค่าหุ้นที่ใช้ค้ำประกัน	บาท
๒๗	๑๐๗-๑๑๒	๖	จำนวนเงินฝากประจำที่ใช้ค้ำประกัน	บาท
๒๘	๑๑๓-๑๒๐	๘	ปี เดือน วันที่อัพเดท	
๒๙	๑๒๑-๑๒๘	๘	(ว่าง)	

ตารางที่ ก.๕ แสดงเขตข้อมูลของระเบียนการค้ำประกันโดยบุคคล

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๖	๖	รหัสประจำตัวสมาชิกผู้ค้ำประกัน	
๒	๗	๑	ลำดับที่ของระเบียนผู้ค้ำประกัน	
๓	๘	๑	รหัสแสดงสถานภาพจริง/ไม่จริงของระเบียน	
๔	๙	๑	รหัสแสดงสภาพถาวร/ชั่วคราวของระเบียน	
๕	๑๐	๑	รหัสแสดงสภาพเก่า/ใหม่ของระเบียน	
๖	๑๑	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๗	๑๒-๑๗	๖	รหัสประจำตัวสมาชิกผู้ขอกู้	
๘	๑๘-๒๒	๕	วงเงินที่ค้ำประกัน	บาท
๙	๒๓-๒๘	๖	รหัสประจำตัวสมาชิกผู้ร่วมค้ำประกัน	
๑๐	๒๙	๑	ลำดับที่ของระเบียนผู้ร่วมค้ำประกัน	
๑๑	๓๐-๓๗	๘	ปี เดือน วันที่อัพเดท	
๑๒	๓๘-๖๔	๒๗	(ว่าง)	

ตารางที่ ก.๖ แสดงเขตข้อมูลของระเบียนการค้ำประกันโดยใช้หลักทรัพย์

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๙	๙	รหัสโฉนดที่ดินที่ใช้ค้ำประกัน	
๒	๑๐	๑	ลำดับที่ของระเบียนหลักทรัพย์ค้ำประกัน	
๓	๑๑	๑	รหัสแสดงสถานภาพจริง/ไม่จริงของระเบียน	
๔	๑๒	๑	รหัสแสดงสถานภาพถาวร/ชั่วคราวของระเบียน	
๕	๑๓	๑	รหัสแสดงสถานภาพเก่า/ใหม่ของระเบียน	
๖	๑๔	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๗	๑๕	๑	รหัสประเภทการกู้	
๘	๑๖-๒๑	๖	รหัสประจำตัวสมาชิกผู้ขอกู้	
๙	๒๒-๒๗	๖	จำนวนต้นเงินกู้	บาท
๑๐	๒๘-๓๓	๖	วงเงินหลักทรัพย์ที่ใช้ค้ำประกัน	บาท
๑๑	๓๔-๔๒	๙	รหัสโฉนดที่ดินที่ใช้ร่วมค้ำประกัน	
๑๒	๔๓	๑	ลำดับที่ของระเบียนหลักทรัพย์ที่ใช้ร่วมค้ำประกัน	
๑๓	๔๔-๕๑	๘	ปี เดือน วันที่อัพเดท	
๑๔	๕๒-๖๔	๑๓	(ว่าง)	

ตารางที่ ก.๗ แสดงเขตข้อมูลของระเบียนรายละเอียดหลักทรัพย์

เขตข้อมูลที่	คอลัมน์ที่	จำนวนคอลัมน์	ชื่อข้อมูล	หน่วย
๑	๑-๔	๔	รหัสโฉนดที่ดิน	
๒	๑๐	๑	รหัสแสดงสถานภาพจริง/ไม่จริงของระเบียน	
๓	๑๑	๑	รหัสแสดงสถานภาพถาวร/ชั่วคราวของระเบียน	
๔	๑๒	๑	รหัสแสดงสถานภาพเก่า/ใหม่ของระเบียน	
๕	๑๓	๑	รหัสแสดงสภาพใช้งาน/ไม่ใช้งานของระเบียน	
๖	๑๔	๑	รหัสประเภทการกู้	
๗	๑๕	๑	จำนวนระเบียนที่ใช้ค้ำประกัน	
๘	๑๖	๑	จำนวนคนที่ร่วมใช้โฉนด	
๙	๑๗-๒๔	๘	ปี เดือน วันที่ประเมินราคา	
๑๐	๒๕-๒๘	๔	จำนวนพื้นที่ที่ดิน	ตารางวา
๑๑	๒๙-๓๔	๖	ราคาประเมินที่ดินต่อตารางวา	บาท
๑๒	๓๕-๔๑	๗	ราคาประเมินที่ดิน	บาท
๑๓	๔๒-๔๘	๗	ราคาประเมินสิ่งปลูกสร้าง	บาท
๑๔	๔๙-๕๔	๖	ราคาประกันไฟของสิ่งปลูกสร้าง	บาท
๑๕	๕๕-๖๐	๖	วงเงินหลักทรัพย์ที่ใช้ค้ำประกัน	บาท
๑๖	๖๑-๖๖	๖	จำนวนเงินหลักทรัพย์คงเหลือ	บาท
๑๗	๖๗-๗๔	๘	ปี เดือน วันที่อัพเดท	
๑๘	๗๕-๑๒๘	๕๔	(ว่าง)	

ภาคผนวก ข.

รหัสที่ใช้แทนข้อมูล

ในการวิจัยนี้ เขตข้อมูลในระเบียนต่าง ๆ จะเก็บข้อมูลแทนด้วยตัวเลขทั้งสิ้นยกเว้น เขตข้อมูลชื่อ นามสกุลภาษาไทย และภาษาอังกฤษเท่านั้นที่แทนด้วยตัวอักษร ซึ่งในภาคผนวกนี้จะอธิบายถึงรหัสที่ใช้แทนข้อมูลในระเบียนต่าง ๆ ตามลำดับ ถ้ามีรหัสใดซ้ำกันจะอธิบายรหัสนั้นในเขตข้อมูลของระเบียนที่กล่าวก่อน

ข.๑ <u>รหัสที่ใช้แทนข้อมูลในระเบียนรายชื่อสมาชิก</u>

<u>เขตข้อมูลที่ ๑</u> เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกแทนด้วยตัวเลข ๖ หลัก สองหลักแรกคือ ปีที่สมัครเข้าเป็นสมาชิก สามหลักต่อไปใช้เป็นเลขลำดับที่ของบุคคลที่สมัครเข้าเป็นสมาชิกในเดือนนั้น และหลักสุดท้ายสร้างขึ้นเพื่อประโยชน์ในการตรวจสอบห้าหลักแรก

<u>เขตข้อมูลที่ ๒</u> เป็นที่เก็บข้อมูลรหัสแสดงสถานภาพที่เป็นจริงหรือไม่เป็นจริงของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	ระเบียนที่เป็นจริง
๑	แทน	ระเบียนที่ไม่เป็นจริง

<u>เขตข้อมูลที่ ๓</u> เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	ระเบียนที่มีสภาพถาวร
๑	แทน	ระเบียนที่มีสภาพชั่วคราว

<u>เขตข้อมูลที่ ๔</u> เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	ระเบียนที่มีสภาพเก่า
๑	แทน	ระเบียนที่มีสภาพใหม่

<u>เขตข้อมูลที่ ๕</u> เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	ระเบียนที่มีสภาพใช้งาน
๑	แทน	ระเบียนที่มีสภาพไม่ใช้งาน

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลเลขประจำตัวสมาชิกเก่า แทนด้วยเลขจำนวนเต็ม ๔ หลัก

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลคำนำหน้าชื่อ ชื่อนามสกุลของสมาชิก แทนด้วยตัวอักษรภาษาไทยหกสิบตัว

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลคำนำหน้าชื่อ ชื่อนามสกุลของสมาชิก แทนด้วยตัวอักษรภาษาอังกฤษสี่สิบตัว

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลรหัสเพศ แทนด้วยเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	เพศชาย
๒	แทน	เพศหญิง

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลรหัสแหล่งจ่ายเงินเดือน แทนด้วยตัวเลข ๓ หลัก รหัสที่ใช้แทนมีดังนี้

๐๐๑	แทน	หน่วยงานในสำนักอธิการบดีที่ไม่สามารถจัดอยู่ในกองใดกองหนึ่งได้
๐๑๐	แทน	กองกลาง
๐๒๐	แทน	กองคลัง
๐๓๐	แทน	กองการเจ้าหน้าที่
๐๔๐	แทน	กองอาคารสถานที่
๐๕๐	แทน	กองบริการการศึกษา
๐๖๐	แทน	กองแผนงาน
๐๗๐	แทน	กองกิจการนิสิต
๐๘๐	แทน	สถาบันวิจัย
๐๙๙	แทน	บัณฑิตวิทยาลัย
๑๐๐-๑๔๙	แทน	คณะอักษรศาสตร์
๑๕๐-๑๙๙	แทน	คณะวิศวกรรมศาสตร์
๒๐๐-๒๔๙	แทน	คณะพาณิชยศาสตร์และการบัญชี
๒๕๐-๒๙๙	แทน	คณะวิทยาศาสตร์

๓๐๐-๓๔๙	แทน	คณะรัฐศาสตร์
๓๕๐-๓๙๙	แทน	คณะสถาปัตยกรรมศาสตร์
๔๐๐-๔๔๙	แทน	คณะครุศาสตร์
๔๕๐-๔๙๙	แทน	คณะแพทยศาสตร์
๕๐๐-๕๔๙	แทน	คณะนิเทศศาสตร์
๕๕๐-๕๙๙	แทน	คณะสัตวแพทยศาสตร์
๖๐๐-๖๔๙	แทน	คณะเศรษฐศาสตร์
๖๕๐-๖๙๙	แทน	คณะทันตแพทยศาสตร์
๗๐๐-๗๔๙	แทน	คณะนิติศาสตร์
๗๕๐-๗๙๙	แทน	คณะเภสัชศาสตร์

<u>เขตข้อมูลที่ ๑๑</u> เป็นที่เก็บข้อมูลรหัสบัญชีถือจ่าย (เลข จ.๑๘) แทนด้วยตัวเลข ๗ หลัก สาม-หลักแรกแทนแผนกวิชาหรือหน่วยงานที่บุคคลากรนั้นสังกัด สามหลักต่อมาแทนเลขที่บัญชีเงินเดือนตามกอง-คลัง และหลักสุดท้ายสร้างขึ้นเพื่อประโยชน์ในการตรวจสอบหกหลักแรก

<u>เขตข้อมูลที่ ๑๒</u> เป็นที่เก็บข้อมูลรหัสประจำตัวข้าราชการแทนด้วยตัวเลข ๗ หลัก สาม-หลักแรกคือรหัสแผนกวิชาหรือหน่วยงานที่สังกัดอยู่ในปัจจุบัน สามหลักต่อมาเป็นลำดับที่ของบุคคลากรในแผนกวิชาหรือหน่วยงานตามลำดับ ตั้งแต่เริ่มเข้าปฏิบัติราชการ ในกรณีที่เข้าปฏิบัติราชการหลายคนในเวลาเดียวกันจะใช้รหัสตามลำดับตัวอักษรไทยของชื่อ และหลักสุดท้ายสร้างขึ้นเพื่อประโยชน์ในการตรวจสอบหกหลักแรก รหัสนี้จะเปลี่ยนแปลงเมื่อย้ายสังกัด

<u>เขตข้อมูลที่ ๑๓</u> เป็นที่เก็บข้อมูลรหัสการเป็นสมาชิกแทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	ไม่เคยเป็นสมาชิกมาก่อน
๑	แทน	เคยเป็นสมาชิกมาก่อน

<u>เขตข้อมูลที่ ๑๔</u> เป็นที่เก็บข้อมูลรหัสเงื่อนไขการเปลี่ยนคำนำหน้าชื่อ ชื่อ นามสกุล แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐ แทน - ไม่มีการเปลี่ยนแปลง

๑ แทน มีการเปลี่ยนคำนำหน้าชื่อ, ชื่อ หรือนามสกุล

๒ แทน เป็นระเบียนของสมาชิกใหม่

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูล ปี เดือน วันที่อัพเดทระเบียนแทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก สี่หลักแรกแทนปีพุทธศักราช สองหลักถัดไปแทนเดือน และสองหลักสุดท้ายแทนวัน

เขตข้อมูลที่ ๑๖ ไม่มีข้อมูลจึงเว้นว่างไว้ ๑๘ หลัก

ข.๒ รหัสที่ใช้แทนข้อมูลในระเบียนสถานภาพของสมาชิก

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิก แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๒ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๓ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๔ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลปีเดือนวันที่เข้าเป็นสมาชิก แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลอายุสมาชิก แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก มีหน่วยเป็นเดือน

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลรหัสแสดงสถานภาพสมรส แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	โสด
๒	แทน	สมรส
๓	แทน	หย่า
๔	แทน	หม้าย
๕	แทน	แยกกันอยู่

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกคู่สมรส แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลรหัสหน่วยงานที่สังกัดเมื่อแรกเข้าทำงาน แทนด้วยตัวเลข ๓ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๑๐ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลรหัสหน่วยงานที่ทำงานในปัจจุบัน แทนด้วยตัวเลข ๓ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๑๐ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลรหัสประเภทสมาชิก แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	ข้าราชการ
๒	แทน	ลูกจ้าง

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลจำนวนเดือนที่เหลือที่จะกู้ใหม่ได้ แทนด้วยตัวเลขจำนวนเต็ม ๒ หลัก มีหน่วยเป็นเดือน

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลวงเงินขอกู้ (สามัญและฉุกเฉิน) แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลรหัสสภาพการกู้ แทนด้วยตัวเลข ๓ หลัก หลักแรกมีรหัสที่ใช้แทนดังนี้

๐	แทน	ไม่มีกู้
๑	แทน	กู้ฉุกเฉิน

หลักที่สองมีรหัสใช้แทนดังนี้

๐	แทน	ไม่มีกู้
๒	แทน	กู้สามัญ

หลักที่สามมีรหัสใช้แทนดังนี้

๐	แทน	ไม่มีกู้
๓	แทน	กู้พิเศษเพื่อเคหะสังเคราะห์
๔	แทน	กู้พิเศษเพื่อซื้อที่ดิน

<u>เขตข้อมูลที่ ๑๖</u> เป็นที่เก็บข้อมูลของเงินค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

<u>เขตข้อมูลที่ ๑๗</u> เป็นที่เก็บข้อมูลจำนวนเงินค้ำประกันที่ใช้แล้ว แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

<u>เขตข้อมูลที่ ๑๘</u> เป็นที่เก็บข้อมูลจำนวนคนที่ค้ำประกันให้ แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก

<u>เขตข้อมูลที่ ๑๙</u> เป็นที่เก็บข้อมูลจำนวนระเบียนที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก

<u>เขตข้อมูลที่ ๒๐</u> เป็นที่เก็บข้อมูลเงินเดือน แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

<u>เขตข้อมูลที่ ๒๑</u> เป็นที่เก็บข้อมูลรายได้อื่นรายเดือน แทนด้วยเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นบาท

<u>เขตข้อมูลที่ ๒๒</u> เป็นที่เก็บข้อมูลเงินสะสมส่งรายเดือนตามจริง แทนด้วยเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นบาท

<u>เขตข้อมูลที่ ๒๓</u> เป็นที่เก็บข้อมูลรายจ่ายอื่นรายเดือนที่หักโดยหน่วยงาน แทนด้วยตัวเลข ๗ หลัก เป็นเลขจำนวนเต็ม ๕ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็น บาท สตางค์

เขตข้อมูลที่ ๒๔ เป็นที่เก็บข้อมูลยอดเงินปรับปรุงเพื่อลดรายจ่ายอื่นรายเดือน แทนด้วยตัวเลข ๕ หลัก เป็นเลขจำนวนเต็ม ๓ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็น บาท สตางค์

เขตข้อมูลที่ ๒๕ เป็นที่เก็บข้อมูลรหัสเงื่อนไขการพ้นสภาพการเป็นสมาชิก แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	ตาย
๒	แทน	วิกลจริต
๓	แทน	ล้มละลาย
๔	แทน	ถูกออกจากราชการ
๕	แทน	ถูกให้ออกจากสหกรณ์ออมทรัพย์
๖	แทน	ลาออกจากสหกรณ์ออมทรัพย์

เขตข้อมูลที่ ๒๖ เป็นที่เก็บข้อมูลปีเดือนวันที่พ้นสภาพการเป็นสมาชิก แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒๗ เป็นที่เก็บข้อมูลปีเดือนวันที่อัพเดท แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒๘ ไม่มีข้อมูลจึงเว้นว่างไว้ ๒๔ หลัก

ข.๓ รหัสที่ใช้แทนข้อมูลในระเบียนเงินสะสม

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิก แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๒ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๓ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๔ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก รหัสนี้ใช้เช่นเดียวกับเขตข้อมูลที่ ๕ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลเงินสะสมรายเดือนตามเกณฑ์ แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลเงินสะสมรายเดือนตามจริง แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลเงินสะสมรายเดือนก่อนการเปลี่ยนแปลง แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูล ปี เดือน ที่เปลี่ยนแปลงการส่งเงินสะสม แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก สี่หลักแรกแทนปี สองหลักสุดท้ายแทนวัน

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลรหัสการเปลี่ยนแปลงเงินสะสม แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	เงินสะสมจริงมากกว่าหรือเท่ากับเงินสะสมตามเกณฑ์
๒	แทน	งดส่งเงินสะสมเนื่องจากครบ ๘๔ งวด
๓	แทน	ของดส่งเพราะเก็บไม่ได้

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลจำนวนครั้งการเปลี่ยนแปลงเงินสะสม แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลรหัสเงื่อนไขการคำนวณยอดเงินค่าหุ้น แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐	แทน	คำนวณยอดเงินค่าหุ้นโดยใช้เงินสะสมจริง
๑	แทน	คำนวณยอดเงินค่าหุ้นโดยใช้เงินสะสมก่อนการเปลี่ยนแปลง

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลยอดเงินค่าหุ้นแทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลยอดเงินที่เพื่อคิดเงินปันผลแทนด้วยตัวเลขจำนวนเต็ม ๗ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลวัน เดือน วันที่อัพเดท แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๖ ไม่มีข้อมูลจึงเว้นว่างไว้ ๑๓ หลัก

ข.๔ รหัสที่ใช้แทนข้อมูลในระเบียนเงินกู้

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิก แทนด้วยตัวเลข ๒ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลรหัสประเภทการกู้ แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๑	แทน	การกู้ฉุกเฉิน
๒	แทน	การกู้สามัญ
๓	แทน	การกู้เพื่อเคหะสงเคราะห์หรืออื่น ๆ
๔	แทน	การกู้พิเศษเพื่อซื้อที่ดินหรือที่ดินพร้อมสิ่งปลูกสร้าง

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพที่เป็นจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวร หรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๓ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลรหัสแสดงภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลรหัสเลขที่สัญญากู้ แทนด้วยตัวเลข ๗ หลัก ห้าหลักแรกเป็นเลขลำดับที่สัญญากู้ สองหลักสุดท้ายเป็นปีที่กู้

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูล ปี เดือน วัน ที่ทำสัญญากู้ แทนด้วยตัวเลข ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูล จำนวนต้นเงินกู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลรหัสการคิดต้นเงินชำระคืน แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก รหัสที่ใช้มีดังนี้

๐ แทน การคิดต้นเงินชำระคืนงวดสุดท้าย

๑ แทน การคิดต้นเงินชำระคืนงวดแรก

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลต้นเงินชำระคืนงวดแรกหรืองวดสุดท้าย แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลต้นเงินชำระคืนงวดคงที่ แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลจำนวนเงินยอดลูกหนี้คงเหลือ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลจำนวนงวดชำระทั้งสิ้น แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลงวดชำระที่ แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก

เขตข้อมูลที่ ๑๖ เป็นที่เก็บข้อมูลดอกเบี้ยชำระงวดแรกหรืองวดสุดท้าย แทนด้วยตัวเลข ๖ หลัก ประกอบด้วย เลขจำนวนเต็ม ๔ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาทสตางค์

เขตข้อมูลที่ ๑๗ เป็นที่เก็บข้อมูลดอกเบี้ยชำระงวดปัจจุบัน แทนด้วยตัวเลข ๖ หลัก ประกอบด้วยเลขจำนวนเต็ม ๔ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาทสตางค์

เขตข้อมูลที่ ๑๘ เป็นที่เก็บข้อมูลส่วนลดดอกเบี้ย แทนด้วยตัวเลข ๕ หลัก ประกอบด้วยตัวเลขจำนวนเต็ม ๓ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาทสตางค์

เขตข้อมูลที่ ๑๙ เป็นที่เก็บข้อมูลอัตราดอกเบี้ยร้อยละต่อปี แทนด้วยตัวเลข ๓ หลัก ประกอบด้วยตัวเลขจำนวนเต็ม ๒ หลัก เลขหลังจุดทศนิยม ๑ หลัก

เขตข้อมูลที่ ๒๐ เป็นที่เก็บข้อมูลดอกเบี้ยสะสม แทนด้วยตัวเลข ๗ หลัก ประกอบด้วยตัวเลขจำนวนเต็ม ๕ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาทสตางค์

เขตข้อมูลที่ ๒๑ เป็นที่เก็บข้อมูล รหัสประจำตัวเลขสมาชิกผู้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒๒ เป็นที่เก็บข้อมูล ลำดับที่ของระเบียนผู้ค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๒๓ เป็นที่เก็บข้อมูล รหัสโฉนดที่ดินที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก หลักแรกแทนรหัสจังหวัด หลักที่สองแทนเขต หลักที่สามแทนอำเภอ สี่หลักต่อมาแทนเล่มที่ของโฉนดที่ดิน และสองหลักสุดท้ายแทนหน้าที่ของโฉนดที่ดิน รหัสจังหวัดที่ใช้แทนมีดังต่อไปนี้

๑	แทน	กรุงเทพมหานคร
๒	แทน	ธนบุรี
๓	แทน	มีนบุรี
๔	แทน	นนทบุรี
๕	แทน	สมุทรปราการ

หลักต่อมาแทนรหัสเขต ซึ่งจังหวัดกรุงเทพมหานครมี ๔ เขต จังหวัดธนบุรีมี ๓ เขต จังหวัดมีนบุรีมี ๑ เขต จังหวัดนนทบุรีมี ๒ เขต และจังหวัดสมุทรปราการมี ๑ เขต

๑๑๐	แทน	จังหวัดกรุงเทพฯ	เขต	๑	อำเภอพระนคร
๑๑๑					อำเภอพระนคร (ดุสิต)
๑๑๒					อำเภอดุสิต
๑๑๓					อำเภอป้อมปราบศัตรูพ่าย (ดุสิต)
๑๑๔					อำเภอพญาไท (ดุสิต)
๑๑๕					อำเภอยานนาวา (บางรัก)
๑๑๖					อำเภอบางรัก
๑๑๗					อำเภอยานนาวา (เมือง)
๑๑๘					อำเภอบางเขน
๑๑๙					อำเภอบางเขน (ตลาดขวัญ)
๑๒๑	แทน	จังหวัดกรุงเทพฯ	เขต	๒	อำเภอดุสิต (บางซื่อ)
๑๒๒					อำเภอบางเขน (บางซื่อ)
๑๒๓					อำเภอบางกะปิ (บางซื่อ)
๑๒๔					อำเภอพญาไท (บางซื่อ)
๑๒๕					อำเภอห้วยขวาง (บางซื่อ)
๑๓๑	แทน	จังหวัดกรุงเทพฯ	เขต	๓	อำเภอพระโขนง
๑๓๒					อำเภอยานนาวา (พระโขนง)
๑๓๓					อำเภอพระโขนง (บางพลีใหญ่)
๑๔๑	แทน	จังหวัดกรุงเทพฯ	เขต	๔	อำเภอบางกะปิ
๑๔๒					อำเภอพญาไท (บางกะปิ)
๑๔๓					อำเภอปทุมวัน (บางกะปิ)
๑๔๔					อำเภอพระโขนง (บางกะปิ)
๑๔๕					อำเภอสัมพันธ์วงศ์ (สามเพ็ง)
๑๔๖					อำเภอปทุมวัน (สามเพ็ง)
๑๔๗					อำเภอป้อมปราบศัตรูพ่าย (สามเพ็ง)
๒๑๑	แทน	จังหวัดธนบุรี	เขต	๑	อำเภอบางขุนเทียน

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

๒๑๒				อำเภอภาษีเจริญ
๒๑๓				อำเภอหนองแขม
๒๒๑	แทน	จังหวัดธนบุรี	เขต ๒	อำเภอตลิ่งชัน
๒๒๒				อำเภอบางกอกน้อย
๒๓๑	แทน	จังหวัดธนบุรี	เขต ๓	อำเภอบางกอกใหญ่
๒๓๒				อำเภอราษฎร์บูรณะ
๒๓๓				อำเภอธนบุรี
๒๓๔				อำเภอคลองสาน
๓๑๑	แทน	จังหวัดมีนบุรี	เขต ๑	อำเภอหนองจอก
๓๑๒				อำเภอลาดกระบัง
๓๑๓				อำเภอมีนบุรี
๔๑๑	แทน	จังหวัดนนทบุรี	เขต ๑	อำเภอเมืองนนทบุรี (ตลาดขวัญ)
๔๒๑	แทน	จังหวัดนนทบุรี	เขต ๒	อำเภอปากเกร็ด
๔๒๒				อำเภอบางกรวย
๔๒๓				อำเภอบางใหญ่
๔๒๔				อำเภอบางบัวทอง
๔๒๕				อำเภอไทรน้อย
๕๑๑	แทน	จังหวัดสมุทรปราการ	เขต ๑	อำเภอเมืองสมุทรปราการ
๕๑๒				อำเภอบางบ่อ
๕๑๓				อำเภอบางพลี
๕๑๔				อำเภอพระประแดง

เขตข้อมูลที่ ๒๔ เป็นที่เก็บข้อมูลลำดับที่ของระเบียนหลักทรัพย์ค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๒๕ เป็นที่เก็บข้อมูลจำนวนโฉนดที่ใช้ค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๒๖ เป็นที่เก็บข้อมูล มูลค่าหุ้นที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๒๗ เป็นที่เก็บข้อมูลจำนวนเงินฝากประจำที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๒๘ เป็นที่เก็บข้อมูล ปีเดือนวันที่อัพเดท แทนด้วยตัวเลขจำนวน ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒๙ ไม่มีข้อมูลจึงเว้นว่างไว้ ๘ หลัก

ข.๕ รหัสที่ใช้แทนข้อมูลในระเบียนการค้ำประกันโดยบุคคล

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกผู้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลลำดับที่ของระเบียนผู้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๑ หลัก

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข จำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลขจำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๓ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลขจำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลขจำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกผู้ขอกู้ แทนด้วยตัวเลข ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลวงเงินที่ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกผู้ร่วมค้ำประกัน แทนด้วยตัวเลข ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลลำดับที่ของระเบียนผู้ร่วมค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลปีเดือนวันที่อัพเดท แทนด้วยตัวเลขจำนวน ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๒ ไม่มีข้อมูลจึงเว้นว่างไว้ ๒๗ หลัก

ข.๖ รหัสที่ใช้แทนเขตข้อมูลในระเบียนการค้ำประกันโดยใช้หลักทรัพย์

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสโฉนดที่ดินที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๔ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๒๓ ในหัวข้อ ข.๔

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลลำดับที่ของระเบียนหลักทรัพย์ค้ำประกัน แทนด้วยตัวเลขจำนวน ๑ หลัก

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๓ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลรหัสประเภทการกู้ แทนด้วยตัวเลขจำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนดังนี้

๓ แทน การกู้พิเศษเพื่อเคหะสังเคราะห์

๔ แทน การกู้พิเศษเพื่อซื้อที่ดิน

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกผู้ขอกู้ แทนด้วยตัวเลข ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลจำนวนต้นเงินกู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลวงเงินหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลรหัสโฉนดที่ดินที่ร่วมใช้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๙ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๒๓ ในหัวข้อ ข.๔

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลลำดับที่ของระเบียนหลักทรัพย์ที่ใช้ร่วมค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลปีเดือนวันที่อัพเดท แทนด้วยตัวเลขจำนวน ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๔ ไม่มีข้อมูลจึงเว้นว่างไว้ ๑๓ หลัก

ข.๗ รหัสที่ใช้แทนเขตข้อมูลในระเบียนรายละเอียดของหลักทรัพย์

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสโฉนดที่ดิน แทนด้วยตัวเลขจำนวน ๙ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๒๓ ในหัวข้อ ข.๔

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลรหัสแสดงสภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๓ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลรหัสประเภทการกู้ แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๗ ในหัวข้อ ข.๖

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลจำนวนระเบียนที่ใช้ค้ำประกัน แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลจำนวนคนที่ร่วมใช้โฉนด แทนด้วยตัวเลข ๑ หลัก

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลปีเดือนวันที่ประเมินราคา แทนด้วยตัวเลขจำนวน ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลจำนวนพื้นที่ที่ดิน แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นตารางวา

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลราคาประเมินที่ดินต่อตารางวา แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลราคาประเมินที่ดิน แทนด้วยตัวเลขจำนวนเต็ม ๗ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลราคาประเมินสิ่งปลูกสร้าง แทนด้วยตัวเลขจำนวนเต็ม ๗ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลราคาประกันไฟของสิ่งปลูกสร้าง แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลวงเงินหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๖ เป็นที่เก็บข้อมูลจำนวนเงินหลักทรัพย์คงเหลือ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๗ เป็นที่เก็บข้อมูลปีเดือนวันที่อัพเดท แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๑๘ ไม่มีข้อมูลจึงเว้นว่างไว้ ๕๔ หลัก

ข.๘ รหัสที่ใช้แทนข้อมูลในระเบียนชั่วคราว

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิก แทนด้วยตัวเลข ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูลรหัสประเภทการกู้ แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๔

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูลรหัสแสดงสถานภาพจริงหรือไม่จริงของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสแสดงสภาพถาวรหรือชั่วคราวของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๓ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสแสดงสภาพเก่าหรือใหม่ของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลรหัสแสดงสภาพใช้งานหรือไม่ใช้งานของระเบียน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๗ ไม่มีข้อมูล จึงเว้นว่างไว้ทั้ง ๗ หลัก

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลปีเดือนวันที่ทำสัญญากู้ แทนด้วยตัวเลข ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลจำนวนต้นเงินกู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลรหัสการคิดต้นเงินชำระคืน แทนด้วยตัวเลข ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับรหัสในเขตข้อมูลที่ ๑๐ ในหัวข้อ ข.๔

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลต้นเงินชำระคืนงวดแรกหรืองวดสุดท้าย แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลต้นเงินชำระคืนงวดคงที่ แทนด้วยตัวเลขจำนวนเต็ม ๕ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลจำนวนเงินยอดลูกหนี้คงเหลือ แทนด้วยตัวเลขจำนวน-เต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลจำนวนงวดชำระทั้งสิ้น แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลงวดชำระที่ แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก

เขตข้อมูลที่ ๑๖ เป็นที่เก็บข้อมูลดอกเบี้ยชำระงวดแรกหรืองวดสุดท้าย แทนด้วยตัวเลขจำนวน ๖ หลัก เป็นเลขจำนวนเต็ม ๔ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาท สตางค์

เขตข้อมูลที่ ๑๗ ไม่มีข้อมูลจึงเว้นว่างไว้ทั้ง ๖ หลัก

เขตข้อมูลที่ ๑๘ เป็นที่เก็บข้อมูลส่วนลดดอกเบี้ย แทนด้วยตัวเลขจำนวน ๕ หลัก เป็นเลขจำนวนเต็ม ๓ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาท สตางค์

เขตข้อมูลที่ ๑๙ เป็นที่เก็บข้อมูลอัตราดอกเบี้ยร้อยละต่อปี แทนด้วยตัวเลขจำนวน ๓ หลัก เป็นจำนวนเต็ม ๒ หลัก เลขหลังจุดทศนิยม ๑ หลัก

เขตข้อมูลที่ ๒๐ เป็นที่เก็บข้อมูลดอกเบี้ยสะสม แทนด้วยตัวเลขจำนวน ๗ หลัก เป็นจำนวน-เต็ม ๕ หลัก เลขหลังจุดทศนิยม ๒ หลัก มีหน่วยเป็นบาท สตางค์

เขตข้อมูลที่ ๒๑ ไม่มีข้อมูล ใช้เลขศูนย์ ๖ หลักแทน

เขตข้อมูลที่ ๒๒ ไม่มีข้อมูล ใช้เลขศูนย์ ๑ หลักแทน

เขตข้อมูลที่ ๒๓ ไม่มีข้อมูล ใช้เลขศูนย์ ๔ หลักแทน

เขตข้อมูลที่ ๒๔ ไม่มีข้อมูล ใช้เลขศูนย์ ๑ หลักแทน

เขตข้อมูลที่ ๒๕ ไม่มีข้อมูล ใช้เลขศูนย์ ๑ หลักแทน

เขตข้อมูลที่ ๒๖ ไม่มีข้อมูล ใช้เลขศูนย์ ๖ หลักแทน

เขตข้อมูลที่ ๒๗ ไม่มีข้อมูล ใช้เลขศูนย์ ๖ หลักแทน

เขตข้อมูลที่ ๒๘ เป็นที่เก็บข้อมูล ปีเดือนวัน ที่อัพเดทของระเบียนแทนด้วยตัวเลขจำนวน ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ในหัวข้อ ข.๑

เขตข้อมูลที่ ๒๙ ไม่มีข้อมูลจึงเว้นว่างไว้ ๘ หลัก

ข.๔ รหัสที่ใช้แทนเขตข้อมูลในระเบียนบัตรจำลอง

เขตข้อมูลที่ ๑ เป็นที่เก็บข้อมูล ปีเดือนที่ขอกู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๔ ในหัวข้อ ข.๓

เขตข้อมูลที่ ๒ เป็นที่เก็บข้อมูล เลขที่แบบพิมพ์คำขอกู้ แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก เรียงลำดับที่ตามเลขที่รับแบบพิมพ์คำขอกู้

เขตข้อมูลที่ ๓ เป็นที่เก็บข้อมูล ลำดับที่ของคนหรือหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก

เขตข้อมูลที่ ๔ เป็นที่เก็บข้อมูลรหัสประจำตัวสมาชิกผู้ขอกู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๕ เป็นที่เก็บข้อมูลรหัสประเภทการกู้ แทนด้วยตัวเลขจำนวน ๑ หลัก โดยมีรหัสที่ใช้แทนเช่นเดียวกับเขตข้อมูลที่ ๒ ของหัวข้อ ข.๔

เขตข้อมูลที่ ๖ เป็นที่เก็บข้อมูลจำนวนต้นเงินที่กู้ แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท

เขตข้อมูลที่ ๗ เป็นที่เก็บข้อมูลจำนวนงวดชำระทั้งสิ้น แทนด้วยตัวเลขจำนวนเต็ม ๓ หลัก

เขตข้อมูลที่ ๘ เป็นที่เก็บข้อมูลรหัสบุคคลหรือหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๔ หลัก ถ้าเป็นบุคคลที่ใช้ค้ำประกัน สามหลักแรกแทนด้วยเลขศูนย์ หกหลักสุดท้ายมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑ ของหัวข้อ ข.๑ ถ้าเป็นหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวน ๔ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๒๓ ของหัวข้อ ข.๔

เขตข้อมูลที่ ๙ เป็นที่เก็บข้อมูลรหัสแสดงสถานะการค้ำประกันของบุคคลหรือหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลข ๑ หลัก รหัสที่ใช้แทนมีดังนี้

๐ แทน ใช้บุคคลหรือหลักทรัพย์เดิมในการค้ำประกัน

๑ แทน ใช้บุคคลหรือหลักทรัพย์ใหม่ในการค้ำประกัน

เขตข้อมูลที่ ๑๐ เป็นที่เก็บข้อมูลจำนวนคนหรือหลักทรัพย์ที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๑ หลัก

เขตข้อมูลที่ ๑๑ เป็นที่เก็บข้อมูลจำนวนเงินที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๖ ตัว

เขตข้อมูลที่ ๑๒ เป็นที่เก็บข้อมูลมูลค่าหุ้นที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๖ ตัว

เขตข้อมูลที่ ๑๓ เป็นที่เก็บข้อมูลจำนวนเงินฝากที่ใช้ค้ำประกัน แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๖ ตัว

เขตข้อมูลที่ ๑๔ เป็นที่เก็บข้อมูลปีเดือนวันที่ประเมินราคาที่ดิน แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ของหัวข้อ ข.๑

เขตข้อมูลที่ ๑๕ เป็นที่เก็บข้อมูลพื้นที่ที่ดินของโฉนด แทนด้วยตัวเลขจำนวนเต็ม ๔ หลัก มีหน่วยเป็นตารางวา ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๔ ตัว

เขตข้อมูลที่ ๑๖ เป็นที่เก็บข้อมูลราคาประเมินที่ดินต่อตารางวา แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๖ ตัว

เขตข้อมูลที่ ๑๗ เป็นที่เก็บข้อมูลราคาประเมินสิ่งก่อสร้าง แทนด้วยตัวเลขจำนวนเต็ม ๗ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๗ ตัว

เขตข้อมูลที่ ๑๘ เป็นที่เก็บข้อมูลราคาประกันไฟสิ่งก่อสร้าง แทนด้วยตัวเลขจำนวนเต็ม ๖ หลัก มีหน่วยเป็นบาท ถ้าไม่มีข้อมูลแทนด้วยเลขศูนย์ ๖ ตัว

เขตข้อมูลที่ ๑๙ เป็นที่เก็บข้อมูล ปีเดือนวันที่อัพเดทของระเบียน แทนด้วยตัวเลขจำนวนเต็ม ๘ หลัก โดยมีวิธีการจัดเรียงข้อมูลเช่นเดียวกับเขตข้อมูลที่ ๑๕ ของหัวข้อ ๘.๑

เขตข้อมูลที่ ๒๐ ไม่มีข้อมูลจึงเว้นว่างไว้ทั้ง ๓๓ หลัก

ข.๑๐ รหัสแสดงสถานะของระเบียน

รหัสแสดงสถานะของระเบียน แทนด้วยเลข ๔ หลัก รหัสและความหมายที่ใช้แทนดังแสดงใน ตารางที่ ข.๑

ตารางที่ ข.๑ แสดงรหัสและความหมายของระเบียน

ลำดับที่	รหัส	ความหมายของระเบียน				หมายเหตุ
		จริง/ไม่จริง	ถาวร/ชั่วคราว	เก่า/ใหม่	ใช้งาน/ไม่ใช้งาน	
๑	๐๐๐๐	จริง	ถาวร	เก่า	ใช้งาน	เกิดขึ้นได้
๒	๐๐๐๑	จริง	ถาวร	เก่า	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้
๓	๐๐๑๐	จริง	ถาวร	ใหม่	ใช้งาน	ไม่สามารถเกิดขึ้นได้
๔	๐๐๑๑	จริง	ถาวร	ใหม่	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้
๕	๐๑๐๐	จริง	ชั่วคราว	เก่า	ใช้งาน	ไม่สามารถเกิดขึ้นได้
๖	๐๑๐๑	จริง	ชั่วคราว	เก่า	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้
๗	๐๑๑๐	จริง	ชั่วคราว	ใหม่	ใช้งาน	เกิดขึ้นได้
๘	๐๑๑๑	จริง	ชั่วคราว	ใหม่	ไม่ใช้งาน	เกิดขึ้นได้
๙	๑๐๐๐	ไม่จริง	ถาวร	เก่า	ใช้งาน	เกิดขึ้นได้
๑๐	๑๐๐๑	ไม่จริง	ถาวร	เก่า	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้
๑๑	๑๐๑๐	ไม่จริง	ถาวร	ใหม่	ใช้งาน	เกิดขึ้นได้
๑๒	๑๐๑๑	ไม่จริง	ถาวร	ใหม่	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้
๑๓	๑๑๐๐	ไม่จริง	ชั่วคราว	เก่า	ใช้งาน	เกิดขึ้นได้
๑๔	๑๑๐๑	ไม่จริง	ชั่วคราว	เก่า	ไม่ใช้งาน	เกิดขึ้นได้
๑๕	๑๑๑๐	ไม่จริง	ชั่วคราว	ใหม่	ใช้งาน	ไม่สามารถเกิดขึ้นได้
๑๖	๑๑๑๑	ไม่จริง	ชั่วคราว	ใหม่	ไม่ใช้งาน	ไม่สามารถเกิดขึ้นได้

ภาคผนวก ค.

โปรแกรมคอมพิวเตอร์ที่ใช้ในการวิจัย

ค.๑ แม็บและดีเซ็คที่ใช้

ค.๑.๑ แม็บและดีเซ็คที่ใช้ในโปรแกรม P01

```
M=MTITLE
         PRINT NOGEN
TITLE    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,DATA=FIELD,TIOAPFX=YES
MTITLE   DFHMDI   SIZE=(24,80),LINE=1,COLUMN=1
         DFHMDF   POS=(4,16),ATTRB=(PROT,BRT),LENGTH=46,                *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
         DFHMDF   POS=(6,25),ATTRB=(PROT,NORM),LENGTH=4,                *
               INITIAL='DATE'
KDATE    DFHMDF   POS=(6,31),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=2,        *
               INITIAL='--'
         DFHMDF   POS=(6,34),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,36),ATTRB=(PROT,BRT),LENGTH=1,                 *
               INITIAL='/'
KMONTH   DFHMDF   POS=(6,38),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,           *
               INITIAL='--'
         DFHMDF   POS=(6,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,43),ATTRB=(PROT,BRT),LENGTH=1,                 *
               INITIAL='/'
KYEAR    DFHMDF   POS=(6,45),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,           *
               INITIAL='----'
         DFHMDF   POS=(6,50),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,26),ATTRB=(PROT,NORM),LENGTH=23,               *
               INITIAL='INFORMATION FOR DISPLAY'
         DFHMDF   POS=(10,23),ATTRB=(PROT,NORM),LENGTH=30,              *
               INITIAL='1 = LOAN DESCRIPTION OF MEMBER'
         DFHMDF   POS=(12,23),ATTRB=(PROT,NORM),LENGTH=25,              *
               INITIAL='2 = STATUS OF MEMBER     '
         DFHMDF   POS=(14,23),ATTRB=(PROT,NORM),LENGTH=32,              *
               INITIAL='3 = DESCRIPTION OF GUARANTEE MAN'
         DFHMDF   POS=(16,23),ATTRB=(PROT,NORM),LENGTH=33,              *
               INITIAL='4 = DESCRIPTION OF GUARANTEE LAND'
         DFHMDF   POS=(19,11),ATTRB=(PROT,BRT),LENGTH=8,                *
               INITIAL='OPTION  '
CHOSEN   DFHMDF   POS=(19,20),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,          *
               INITIAL='-'
         DFHMDF   POS=(19,22),ATTRB=(ASKIP),LENGTH=1
ERR      DFHMDF   POS=(23,20),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(23,61),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
D=C.MTITLE
         PRINT NOGEN
TITLE    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,DATA=FIELD,TIOAPFX=YES
MTITLE   DFHMDI   SIZE=(24,80),LINE=1,COLUMN=1
         DFHMDF   POS=(4,16),ATTRB=(PROT,BRT),LENGTH=46,               *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
         DFHMDF   POS=(6,25),ATTRB=(PROT,NORM),LENGTH=4,               *
               INITIAL='DATE'
KDATE    DFHMDF   POS=(6,31),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=2,       *
               INITIAL='--'
         DFHMDF   POS=(6,34),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,36),ATTRB=(PROT,BRT),LENGTH=1,                *
               INITIAL='/'
KMONTH   DFHMDF   POS=(6,38),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,          *
               INITIAL='--'
         DFHMDF   POS=(6,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,43),ATTRB=(PROT,BRT),LENGTH=1,                *
               INITIAL='/'
KYEAR    DFHMDF   POS=(6,45),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,          *
               INITIAL='----'
         DFHMDF   POS=(6,50),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,26),ATTRB=(PROT,NORM),LENGTH=23,              *
               INITIAL='INFORMATION FOR DISPLAY'
         DFHMDF   POS=(10,23),ATTRB=(PROT,NORM),LENGTH=30,             *
               INITIAL='1 = LOAN DESCRIPTION OF MEMBER'
         DFHMDF   POS=(12,23),ATTRB=(PROT,NORM),LENGTH=25,             *
               INITIAL='2 = STATUS OF MEMBER     '
         DFHMDF   POS=(14,23),ATTRB=(PROT,NORM),LENGTH=32,             *
               INITIAL='3 = DESCRIPTION OF GUARANTEE MAN'
         DFHMDF   POS=(16,23),ATTRB=(PROT,NORM),LENGTH=33,             *
               INITIAL='4 = DESCRIPTION OF GUARANTEE LAND'
         DFHMDF   POS=(19,11),ATTRB=(PROT,BRT),LENGTH=8,               *
               INITIAL='OPTION  '
CHOSEN   DFHMDF   POS=(19,20),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,         *
               INITIAL='-'
         DFHMDF   POS=(19,22),ATTRB=(ASKIP),LENGTH=1
LERR     DFHMDF   POS=(23,20),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(23,61),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
M=MAPERR1

          PRINT NOGEN
CMAP      DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
                   CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1   DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
ERR       DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,              *
                   JUSTIFY=(LEFT,BLANK)
          DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
          DFHMSD   TYPE=FINAL
          END
```

D=C.MAPERR1

```
         PRINT NOGEN
CMAP     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1  DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
LERR     DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,             *
               JUSTIFY=(LEFT,BLANK)
         DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

ค.๑.๒ แม็บและดีเซ็คที่ใช้ในโปรแกรม P02

```
MAPS
         PRINT NOGEN
LPAPI    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TIOA=STD,
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LPAPF    DFHMDI   SIZE=(24,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=42,
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
KDATE    DFHMDF   POS=(1,66),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,
               INITIAL='LOAN DESCRIPTION'
         DFHMDF   POS=(3,51),ATTRB=(PROT,NORM),LENGTH=11,
               INITIAL='ID-PERSON 7'
KEYLN    DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,ALP,IC),LENGTH=6,
               INITIAL='------'
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,
               INITIAL='PAPE'
KNAME    DFHMDF   POS=(3,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(3,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
B=C.LMAPP

         PRINT NOGEN
LMAPI    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TIOA=STD,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LMAPF    DFHMDI   SIZE=(3,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                *
               INITIAL='SAVING COOPERATIVE OF CHILLICOTHE OHIO UNIVERSITY'
KDATE    DFHMDF   POS=(1,66),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,               *
               INITIAL='LOAN DESCRIPTION'
         DFHMDF   POS=(2,5),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='ID-NUMBER ?'
KEYLN    DFHMDF   POS=(2,23),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=6,        *
               INITIAL='------'
         DFHMDF   POS=(2,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=4,                *
               INITIAL='NAME'
KNAME    DFHMDF   POS=(2,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(2,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
*=LODESCT
         PRINT NOGEN
LODESL   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LODESCT  DFHMDI   SIZE=(15,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                 *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
LDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,                *
               INITIAL='LOAN DESCRIPTION'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER  '
KIDMEM   DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='NAME'
NAMED    DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(3,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,9),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='ID-SPOUSE'
IDMARY   DFHMDF   POS=(4,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(4,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,7),ATTRB=(PROT,NORM),LENGTH=3,                  *
               INITIAL='L/T'
         DFHMDF   POS=(6,12),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='PRINCIPAL'
         DFHMDF   POS=(6,23),ATTRB=(PROT,NORM),LENGTH=7,                 *
               INITIAL='PERIOD#'
         DFHMDF   POS=(6,32),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='DEBT-REMAIN'
         DFHMDF   POS=(6,46),ATTRB=(PROT,NORM),LENGTH=6,                 *
               INITIAL='GUA-ID'
         DFHMDF   POS=(6,56),ATTRB=(PROT,NORM),LENGTH=7,                 *
               INITIAL='AMT-GUA'
         DFHMDF   POS=(6,67),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='AMT-REMAIN'
TORP1    DFHMDF   POS=(8,1),ATTRB=(UNPROT,BRT),LENGTH=4
         DFHMDF   POS=(8,6),ATTRB=(ASKIP),LENGTH=1
OTYPE1   DFHMDF   POS=(8,8),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(8,10),ATTRB=(ASKIP),LENGTH=1
OPRINC1  DFHMDF   POS=(8,14),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(8,20),ATTRB=(ASKIP),LENGTH=1
OPERID1  DFHMDF   POS=(8,25),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,            *
               PICOUT='9'
         DFHMDF   POS=(8,27),ATTRB=(ASKIP),LENGTH=1
ODEBT1   DFHMDF   POS=(8,34),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(8,40),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF   POS=(9,1),ATTRB=(UNPROT,BRT),LENGTH=4
         DFHMDF   POS=(9,6),ATTRB=(ASKIP),LENGTH=1
OTYPE    DFHMDF   POS=(9,8),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(9,10),ATTRB=(ASKIP),LENGTH=1
OPRINC   DFHMDF   POS=(9,12),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(9,20),ATTRB=(ASKIP),LENGTH=1
OPERID   DFHMDF   POS=(9,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,            *
               PICOUT='ZZ9'
         DFHMDF   POS=(9,27),ATTRB=(ASKIP),LENGTH=1
ODEBT    DFHMDF   POS=(9,32),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(9,40),ATTRB=(ASKIP),LENGTH=1
OGUA1    DFHMDF   POS=(9,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(9,53),ATTRB=(ASKIP),LENGTH=1
OAMTG1   DFHMDF   POS=(9,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(9,63),ATTRB=(ASKIP),LENGTH=1
OREM1    DFHMDF   POS=(9,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(9,76),ATTRB=(ASKIP),LENGTH=1
OGUA2    DFHMDF   POS=(10,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               PICOUT='ZZZ999999'
```

```
         DFHMDF   POS=(10,53),ATTRB=(ASKIP),LENGTH=1
OAMTG2   DFHMDF   POS=(10,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(10,63),ATTRB=(ASKIP),LENGTH=1
OREM2    DFHMDF   POS=(10,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,76),ATTRB=(ASKIP),LENGTH=1
OGUA3    DFHMDF   POS=(11,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(11,53),ATTRB=(ASKIP),LENGTH=1
OAMTG3   DFHMDF   POS=(11,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(11,63),ATTRB=(ASKIP),LENGTH=1
OREM3    DFHMDF   POS=(11,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,76),ATTRB=(ASKIP),LENGTH=1
OGUA4    DFHMDF   POS=(12,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(12,53),ATTRB=(ASKIP),LENGTH=1
OAMTG4   DFHMDF   POS=(12,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(12,63),ATTRB=(ASKIP),LENGTH=1
OREM4    DFHMDF   POS=(12,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,76),ATTRB=(ASKIP),LENGTH=1
OGUA5    DFHMDF   POS=(13,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(13,53),ATTRB=(ASKIP),LENGTH=1
OAMTG5   DFHMDF   POS=(13,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(13,63),ATTRB=(ASKIP),LENGTH=1
OREM5    DFHMDF   POS=(13,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,76),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(14,25),ATTRB=(PROT,NORM),LENGTH=16,                *
               INITIAL='SHARE OR DEPOSIT'
SDEP1    DFHMDF   POS=(14,45),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,            *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,53),ATTRB=(ASKIP),LENGTH=1
SDEP     DFHMDF   POS=(14,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,63),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(15,9),ATTRB=(PROT,NORM),LENGTH=15,                 *
               INITIAL='PREVIOUS DEBT '
PRED     DFHMDF   POS=(15,26),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(15,28),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(15,31),ATTRB=(PROT,NORM),LENGTH=17,                *
               INITIAL='(Y = YES, N = NO)'
         DFHMSD   TYPE=FINAL
                  END
```

```
D=C.LDESCT
         PRINT NOGEN
LDDESI   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LDESCT   DFHMDI   SIZE=(15,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                  *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
LDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,                 *
               INITIAL='LOAN DESCRIPTION'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,                  *
               INITIAL='ID-MEMBER  '
KIDMEM   DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,                  *
               INITIAL='NAME'
NAMEO    DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(3,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,9),ATTRB=(PROT,NORM),LENGTH=9,                   *
               INITIAL='ID-SPOUSE'
IDMARY   DFHMDF   POS=(4,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(4,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,7),ATTRB=(PROT,NORM),LENGTH=3,                   *
               INITIAL='L/T'
         DFHMDF   POS=(6,12),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='PRINCIPAL'
         DFHMDF   POS=(6,23),ATTRB=(PROT,NORM),LENGTH=7,                  *
               INITIAL='PERIOD#'
         DFHMDF   POS=(6,32),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='DEBT-REMAIN'
         DFHMDF   POS=(6,46),ATTRB=(PROT,NORM),LENGTH=6,                  *
               INITIAL='GUA-ID'
         DFHMDF   POS=(6,56),ATTRB=(PROT,NORM),LENGTH=7,                  *
               INITIAL='AMT-GUA'
         DFHMDF   POS=(6,67),ATTRB=(PROT,NORM),LENGTH=10,                 *
               INITIAL='AMT-REMAIN'
TORP1    DFHMDF   POS=(8,1),ATTRB=(UNPROT,BRT),LENGTH=4
         DFHMDF   POS=(8,6),ATTRB=(ASKIP),LENGTH=1
OTYPE1   DFHMDF   POS=(8,8),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(8,10),ATTRB=(ASKIP),LENGTH=1
OPRINC1  DFHMDF   POS=(8,14),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,             *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(8,20),ATTRB=(ASKIP),LENGTH=1
OPERID1  DFHMDF   POS=(8,25),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,             *
               PICOUT='9'
         DFHMDF   POS=(8,27),ATTRB=(ASKIP),LENGTH=1
ODEBT1   DFHMDF   POS=(8,34),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,             *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(8,40),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF   POS=(9,1),ATTRB=(UNPROT,BRT),LENGTH=4
         DFHMDF   POS=(9,6),ATTRB=(ASKIP),LENGTH=1
OTYPE    DFHMDF   POS=(9,8),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(9,10),ATTRB=(ASKIP),LENGTH=1
OPRINC   DFHMDF   POS=(9,12),ATTRB=(UNPROT,BRT),LENGTH=7,                 *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(9,20),ATTRB=(ASKIP),LENGTH=1
OPERID   DFHMDF   POS=(9,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,             *
               PICOUT='Z99'
         DFHMDF   POS=(9,27),ATTRB=(ASKIP),LENGTH=1
ODEBT    DFHMDF   POS=(9,32),ATTRB=(UNPROT,BRT),LENGTH=7,                 *
               PICOUT='ZZZ,Z99'
         DFHMDF   POS=(9,40),ATTRB=(ASKIP),LENGTH=1
OGUA1    DFHMDF   POS=(9,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,             *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(9,53),ATTRB=(ASKIP),LENGTH=1
OAMTG1   DFHMDF   POS=(9,55),ATTRB=(UNPROT,BRT),LENGTH=7,                 *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(9,63),ATTRB=(ASKIP),LENGTH=1
OREM1    DFHMDF   POS=(9,68),ATTRB=(UNPROT,BRT),LENGTH=7,                 *
               PICOUT='ZZZ,Z99'
         DFHMDF   POS=(9,76),ATTRB=(ASKIP),LENGTH=1
OGUA2    DFHMDF   POS=(10,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
```

```
         DFHMDF   POS=(10,53),ATTRB=(ASKIP),LENGTH=1
OAMTG2   DFHMDF   POS=(10,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(10,63),ATTRB=(ASKIP),LENGTH=1
OREM2    DFHMDF   POS=(10,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,76),ATTRB=(ASKIP),LENGTH=1
OGUA3    DFHMDF   POS=(11,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(11,53),ATTRB=(ASKIP),LENGTH=1
OAMTG3   DFHMDF   POS=(11,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(11,63),ATTRB=(ASKIP),LENGTH=1
OREM3    DFHMDF   POS=(11,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,76),ATTRB=(ASKIP),LENGTH=1
OGUA4    DFHMDF   POS=(12,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(12,53),ATTRB=(ASKIP),LENGTH=1
OAMTG4   DFHMDF   POS=(12,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(12,63),ATTRB=(ASKIP),LENGTH=1
OREM4    DFHMDF   POS=(12,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,76),ATTRB=(ASKIP),LENGTH=1
OGUA5    DFHMDF   POS=(13,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZZ999999'
         DFHMDF   POS=(13,53),ATTRB=(ASKIP),LENGTH=1
OAMTG5   DFHMDF   POS=(13,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(13,63),ATTRB=(ASKIP),LENGTH=1
OREM5    DFHMDF   POS=(13,68),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,76),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(14,25),ATTRB=(PROT,NORM),LENGTH=16,                *
               INITIAL='SHARE OR DEPOSIT'
SDEP1    DFHMDF   POS=(14,45),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,            *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,53),ATTRB=(ASKIP),LENGTH=1
SDEP     DFHMDF   POS=(14,55),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,63),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(15,9),ATTRB=(PROT,NORM),LENGTH=15,                 *
               INITIAL='PREVIOUS DEBT '
PRED     DFHMDF   POS=(15,26),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(15,28),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(15,31),ATTRB=(PROT,NORM),LENGTH=17,                *
               INITIAL='(Y = YES, N = NO)'
         DFHMSD   TYPE=FINAL
                  END
```

```
#=INQYL

         PRINT NOGEN
MLODES   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
INQYL    DFHMDI   SIZE=(5,80),LINE=16,COLUMN=1
LINE1    DFHMDF   POS=(01,2),ATTRB=(UNPROT,BRT),LENGTH=75
         DFHMDF   POS=(01,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,9),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='LOAN TYPE'
NTYPE    DFHMDF   POS=(2,20),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=1,         *
               INITIAL='-'
         DFHMDF   POS=(2,22),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,25),ATTRB=(PROT,NORM),LENGTH=38,                *
               INITIAL='( 1 = EMER,2 = ORDI,3 = MORT,4 = LAND,'
         DFHMDF   POS=(2,64),ATTRB=(PROT,BRT),LENGTH=7,                  *
               INITIAL='9 = END'
         DFHMDF   POS=(2,72),ATTRB=(PROT,NORM),LENGTH=1,                 *
               INITIAL=')'
NPRINC   DFHMDF   POS=(03,11),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,           *
               INITIAL='------'
         DFHMDF   POS=(03,18),ATTRB=(ASKIP),LENGTH=1
NPERIO   DFHMDF   POS=(03,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,           *
               INITIAL='---'
         DFHMDF   POS=(03,27),ATTRB=(ASKIP),LENGTH=1
NGUA     DFHMDF   POS=(03,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               INITIAL='---------'
         DFHMDF   POS=(03,53),ATTRB=(ASKIP),LENGTH=1
NAMTG    DFHMDF   POS=(03,56),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,           *
               INITIAL='------'
         DFHMDF   POS=(03,63),ATTRB=(ASKIP),LENGTH=1
NREM     DFHMDF   POS=(03,69),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='++++++9'
         DFHMDF   POS=(03,76),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(04,10),ATTRB=(PROT,NORM),LENGTH=20,               *
               INITIAL='(PRINCIPAL) (PERIOD)'
         DFHMDF   POS=(04,44),ATTRB=(PROT,NORM),LENGTH=8,                *
               INITIAL='(GUA-ID)'
         DFHMDF   POS=(04,56),ATTRB=(PROT,NORM),LENGTH=9,                *
               INITIAL='(AMT-GUA)'
         DFHMDF   POS=(04,67),ATTRB=(PROT,NORM),LENGTH=12,               *
               INITIAL='(AMT-REMAIN)'
         DFHMSD   TYPE=FINAL
                  END
```

```
DEC.INQYL

                PRINT NOGEN
MODES           DFHMSD  TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
                      CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
INQYL           DFHMDI  SIZE=(5,80),LINE=16,COLUMN=1
LINE1           DFHMDF  POS=(01,2),ATTRB=(UNPROT,BRT),LENGTH=75
                DFHMDF  POS=(01,79),ATTRB=(ASKIP),LENGTH=1
                DFHMDF  POS=(2,9),ATTRB=(PROT,NORM),LENGTH=9,          *
                      INITIAL='LOAN TYPE'
NTYPE           DFHMDF  POS=(2,20),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=1,  *
                      INITIAL='-'
                DFHMDF  POS=(2,22),ATTRB=(ASKIP),LENGTH=1
                DFHMDF  POS=(2,25),ATTRB=(PROT,NORM),LENGTH=38,        *
                      INITIAL='( 1 = EMER,2 = ORDI,3 = MORT,4 = LAND,'
                DFHMDF  POS=(2,64),ATTRB=(PROT,BRT),LENGTH=7,          *
                      INITIAL='9 = END'
                DFHMDF  POS=(2,72),ATTRB=(PROT,NORM),LENGTH=1,         *
                      INITIAL=')'
NPRINC          DFHMDF  POS=(03,11),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,   *
                      INITIAL='------'
                DFHMDF  POS=(03,18),ATTRB=(ASKIP),LENGTH=1
NPERIO          DFHMDF  POS=(03,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,   *
                      INITIAL='---'
                DFHMDF  POS=(03,27),ATTRB=(ASKIP),LENGTH=1
NGUA            DFHMDF  POS=(03,43),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,   *
                      INITIAL='---------'
                DFHMDF  POS=(03,53),ATTRB=(ASKIP),LENGTH=1
NAMTG           DFHMDF  POS=(03,56),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,   *
                      INITIAL='------'
                DFHMDF  POS=(03,63),ATTRB=(ASKIP),LENGTH=1
NREM            DFHMDF  POS=(03,68),ATTRB=(UNPROT,BRT),LENGTH=7,       *
                      PICOUT='+++++9'
                DFHMDF  POS=(03,76),ATTRB=(ASKIP),LENGTH=1
                DFHMDF  POS=(04,10),ATTRB=(PROT,NORM),LENGTH=20,       *
                      INITIAL='(PRINCIPAL) (PERIOD)'
                DFHMDF  POS=(04,44),ATTRB=(PROT,NORM),LENGTH=9,        *
                      INITIAL='(GUA-ID)'
                DFHMDF  POS=(04,56),ATTRB=(PROT,NORM),LENGTH=9,        *
                      INITIAL='(AMT-GUA)'
                DFHMDF  POS=(04,67),ATTRB=(PROT,NORM),LENGTH=12,       *
                      INITIAL='(AMT-REMAIN)'
                DFHMSD  TYPE=FINAL
                        END
```

```
M=MAPERR1

          PRINT NOGEN
CMAP      DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1   DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
ERR       DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,              *
               JUSTIFY=(LEFT,BLANK)
          DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
          DFHMSD   TYPE=FINAL
          END
```

```
D=C.MAPERR1

          PRINT NOGEN
CMAP      DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1   DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
LERR      DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,            *
               JUSTIFY=(LEFT,BLANK)
          DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
          DFHMSD   TYPE=FINAL
          END
```

ค.๑.๓ แม็บและดีเซ็คที่ใช้ในโปรแกรม P03

```
M=MAP
         PRINT GEN
VARIS    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,DATA=FIELD,TIOAPFX=YES
MAP      DFHMDI   SIZE=(3,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,               *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
KDATE    DFHMDF   POS=(1,69),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,36),ATTRB=(PROT,NORM),LENGTH=11,              *
               INITIAL='LOAN STATUS'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,               *
               INITIAL='ID-MEMBER ?'
KEYLN    DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=6,       *
               INITIAL='------'
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,               *
               INITIAL='NAME'
KNAME    DFHMDF   POS=(3,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(3,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.IMAP
         PRINT NOGEN
MAPIS    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=FREEKB,DATA=FIELD,TIOAPFX=YES
*MAP     DFHMDI   SIZE=(3,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                 *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
KDATE    DFHMDF   POS=(1,69),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,36),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN STATUS'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER ?'
KEYLN    DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=6,         *
               INITIAL='------'
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='NAME'
KNAME    DFHMDF   POS=(3,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(3,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
MAPLOSTAT
         PRINT NOGEN
LOSTAT   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,STORAGE=AUTO,TIOAPFX=YES
CLOSTAT  DFHMDI   SIZE=(13,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                 *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
PDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,36),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN STATUS'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER ?'
KEYIDME  DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='NAME'
NAME1    DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(5,9),ATTRB=(PROT,NORM),LENGTH=4,                  *
               INITIAL='TYPE'
         DFHMDF   POS=(5,20),ATTRB=(PROT,NORM),LENGTH=3,                 *
               INITIAL='AGE'
         DFHMDF   POS=(5,31),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='INCM'
         DFHMDF   POS=(5,42),ATTRB=(PROT,NORM),LENGTH=3,                 *
               INITIAL='NET'
         DFHMDF   POS=(5,53),ATTRB=(PROT,NORM),LENGTH=5,                 *
               INITIAL='SHARE'
         DFHMDF   POS=(5,64),ATTRB=(PROT,NORM),LENGTH=8,                 *
               INITIAL='XF(PRES)'
MEMTYPE  DFHMDF   POS=(7,9),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(7,11),ATTRB=(ASKIP),LENGTH=1
MEMAGE   DFHMDF   POS=(7,18),ATTRB=(UNPROT,BRT),LENGTH=5
         DFHMDF   POS=(7,24),ATTRB=(ASKIP),LENGTH=1
INCOME   DFHMDF   POS=(7,28),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,35),ATTRB=(ASKIP),LENGTH=1
NINCOME  DFHMDF   POS=(7,39),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,46),ATTRB=(ASKIP),LENGTH=1
TOSHARE  DFHMDF   POS=(7,50),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(7,58),ATTRB=(ASKIP),LENGTH=1
XFPRES   DFHMDF   POS=(7,63),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,70),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,4),ATTRB=(PROT,BRT),LENGTH=14,                  *
               INITIAL='**LOAN RIGHT**'
         DFHMDF   POS=(9,9),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='LOAN TYPE'
         DFHMDF   POS=(9,26),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='LOAN RIGHT'
         DFHMDF   POS=(9,43),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN/PERIOD'
         DFHMDF   POS=(9,59),ATTRB=(PROT,NORM),LENGTH=8,                 *
               INITIAL='INTEREST'
         DFHMDF   POS=(9,71),ATTRB=(PROT,NORM),LENGTH=6,                 *
               INITIAL='PERIOD'
LOTYPE1  DFHMDF   POS=(11,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(11,14),ATTRB=(ASKIP),LENGTH=1
LORATE1  DFHMDF   POS=(11,28),ATTRB=(UNPROT,BRT),LENGTH=5,               *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(11,34),ATTRB=(ASKIP),LENGTH=1
LPERIOD1 DFHMDF   POS=(11,45),ATTRB=(UNPROT,BRT),LENGTH=5,               *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(11,51),ATTRB=(ASKIP),LENGTH=1
INTE     DFHMDF   POS=(11,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(11,67),ATTRB=(ASKIP),LENGTH=1
PERIOD1  DFHMDF   POS=(11,73),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,           *
               PICOUT='9'
         DFHMDF   POS=(11,75),ATTRB=(ASKIP),LENGTH=1
LOTYPE2  DFHMDF   POS=(12,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(12,14),ATTRB=(ASKIP),LENGTH=1
LORATE2  DFHMDF   POS=(12,27),ATTRB=(UNPROT,BRT),LENGTH=6,               *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(12,34),ATTRB=(ASKIP),LENGTH=1
```

```
LPERID2  DFHMDF  POS=(12,44),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(12,51),ATTRB=(ASKIP),LENGTH=1
INTOR    DFHMDF  POS=(12,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF  POS=(12,67),ATTRB=(ASKIP),LENGTH=1
PERIOD2  DFHMDF  POS=(12,72),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,           *
               PICOUT='29'
         DFHMDF  POS=(12,75),ATTRB=(ASKIP),LENGTH=1
LOTYPE3  DFHMDF  POS=(13,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF  POS=(13,14),ATTRB=(ASKIP),LENGTH=1
LORATE3  DFHMDF  POS=(13,26),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(13,34),ATTRB=(ASKIP),LENGTH=1
LPERID3  DFHMDF  POS=(13,44),ATTRB=(UNPROT,BRT),LENGTH=6,               *
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(13,51),ATTRB=(ASKIP),LENGTH=1
INTEX    DFHMDF  POS=(13,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF  POS=(13,67),ATTRB=(ASKIP),LENGTH=1
PERIOD3  DFHMDF  POS=(13,71),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,           *
               PICOUT='ZZ9'
         DFHMDF  POS=(13,75),ATTRB=(ASKIP),LENGTH=1
OLDBT    DFHMDI  SIZE=(6,80),LINE=14,COLUMN=1
         DFHMDF  POS=(01,4),ATTRB=(PROT,BRT),LENGTH=12,                 *
               INITIAL='**OLD DEBT**'
         DFHMDF  POS=(02,9),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LOAN TYPE'
         DFHMDF  POS=(02,26),ATTRB=(PROT,NORM),LENGTH=9,                *
               INITIAL='PRINCIPAL'
         DFHMDF  POS=(02,43),ATTRB=(PROT,NORM),LENGTH=11,               *
               INITIAL='DEBT REMAIN'
         DFHMDF  POS=(02,61),ATTRB=(PROT,NORM),LENGTH=7,                *
               INITIAL='PERIOD'
OLOL1    DFHMDF  POS=(03,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF  POS=(03,14),ATTRB=(ASKIP),LENGTH=1
PRINCE1  DFHMDF  POS=(03,28),ATTRB=(UNPROT,BRT),LENGTH=5,               *
               PICOUT='Z,ZZ9'
         DFHMDF  POS=(03,34),ATTRB=(ASKIP),LENGTH=1
DEBTRE1  DFHMDF  POS=(03,43),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(03,51),ATTRB=(ASKIP),LENGTH=1
FPERID1  DFHMDF  POS=(03,61),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,           *
               PICOUT='ZZ9'
         DFHMDF  POS=(03,65),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF  POS=(03,67),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(03,79),ATTRB=(ASKIP),LENGTH=1
OLOL2    DFHMDF  POS=(04,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF  POS=(04,14),ATTRB=(ASKIP),LENGTH=1
PRINCE2  DFHMDF  POS=(04,26),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(04,34),ATTRB=(ASKIP),LENGTH=1
DEBTRE2  DFHMDF  POS=(04,43),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(04,51),ATTRB=(ASKIP),LENGTH=1
FPERID2  DFHMDF  POS=(04,61),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,           *
               PICOUT='ZZ9'
         DFHMDF  POS=(04,65),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF  POS=(04,67),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(04,79),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF  POS=(05,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF  POS=(05,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(06,19),ATTRB=(PROT,NORM),LENGTH=15,               *
               INITIAL='PREVIOUS DEBT '
KDEBT    DFHMDF  POS=(06,36),ATTRB=(UNPROT,BRT,IC),LENGTH=1,            *
               INITIAL='-'
         DFHMDF  POS=(06,38),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(06,41),ATTRB=(PROT,NORM),LENGTH=16,               *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD  TYPE=FINAL
         END
```

```
D=C.OLOSTAT
         PRINT NOGEN
LOSTAT   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,STORAGE=AUTO,TIOAPFX=YES
CLOSTAT  DFHMDI   SIZE=(13,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                 *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
PDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,NORM),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,36),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN STATUS'
         DFHMDF   POS=(3,9),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER ?'
KEYIDME  DFHMDF   POS=(3,23),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(3,30),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,33),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='NAME'
NAME1    DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(5,9),ATTRB=(PROT,NORM),LENGTH=4,                  *
               INITIAL='TYPE'
         DFHMDF   POS=(5,20),ATTRB=(PROT,NORM),LENGTH=3,                 *
               INITIAL='AGE'
         DFHMDF   POS=(5,31),ATTRB=(PROT,NORM),LENGTH=4,                 *
               INITIAL='INCM'
         DFHMDF   POS=(5,42),ATTRB=(PROT,NORM),LENGTH=3,                 *
               INITIAL='NET'
         DFHMDF   POS=(5,53),ATTRB=(PROT,NORM),LENGTH=5,                 *
               INITIAL='SHARE'
         DFHMDF   POS=(5,64),ATTRB=(PROT,NORM),LENGTH=8,                 *
               INITIAL='XF(PRES)'
MEMTYPE  DFHMDF   POS=(7,9),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(7,11),ATTRB=(ASKIP),LENGTH=1
MEMAGE   DFHMDF   POS=(7,18),ATTRB=(UNPROT,BRT),LENGTH=5
         DFHMDF   POS=(7,24),ATTRB=(ASKIP),LENGTH=1
INCOME   DFHMDF   POS=(7,28),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,35),ATTRB=(ASKIP),LENGTH=1
NINCOME  DFHMDF   POS=(7,39),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,46),ATTRB=(ASKIP),LENGTH=1
TOSHARE  DFHMDF   POS=(7,50),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(7,58),ATTRB=(ASKIP),LENGTH=1
XFPRES   DFHMDF   POS=(7,63),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(7,70),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,4),ATTRB=(PROT,BRT),LENGTH=14,                  *
               INITIAL='**LOAN RIGHT**'
         DFHMDF   POS=(9,9),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='LOAN TYPE'
         DFHMDF   POS=(9,26),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='LOAN RIGHT'
         DFHMDF   POS=(9,43),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN/PERIOD'
         DFHMDF   POS=(9,59),ATTRB=(PROT,NORM),LENGTH=8,                 *
               INITIAL='INTEREST'
         DFHMDF   POS=(9,71),ATTRB=(PROT,NORM),LENGTH=6,                 *
               INITIAL='PERIOD'
LOTYPE1  DFHMDF   POS=(11,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(11,14),ATTRB=(ASKIP),LENGTH=1
LORATE1  DFHMDF   POS=(11,28),ATTRB=(UNPROT,BRT),LENGTH=5,               *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(11,34),ATTRB=(ASKIP),LENGTH=1
LPERIOD1 DFHMDF   POS=(11,45),ATTRB=(UNPROT,BRT),LENGTH=5,               *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(11,51),ATTRB=(ASKIP),LENGTH=1
INTE     DFHMDF   POS=(11,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,           *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(11,67),ATTRB=(ASKIP),LENGTH=1
PERIOD1  DFHMDF   POS=(11,73),ATTRB=(UNPROT,BRT,NUM),LENGTH=1,           *
               PICOUT='9'
         DFHMDF   POS=(11,75),ATTRB=(ASKIP),LENGTH=1
LOTYPE2  DFHMDF   POS=(12,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(12,14),ATTRB=(ASKIP),LENGTH=1
LORATE2  DFHMDF   POS=(12,27),ATTRB=(UNPROT,BRT),LENGTH=6,               *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(12,34),ATTRB=(ASKIP),LENGTH=1
```

```
LPERIOD2 DFHMDF   POS=(12,44),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(12,51),ATTRB=(ASKIP),LENGTH=1
INTOR    DFHMDF   POS=(12,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(12,67),ATTRB=(ASKIP),LENGTH=1
PERIOD2  DFHMDF   POS=(12,72),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,            *
               PICOUT='Z9'
         DFHMDF   POS=(12,75),ATTRB=(ASKIP),LENGTH=1
LOTYPE3  DFHMDF   POS=(13,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(13,14),ATTRB=(ASKIP),LENGTH=1
LORATE3  DFHMDF   POS=(13,26),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,34),ATTRB=(ASKIP),LENGTH=1
LPERID3  DFHMDF   POS=(13,44),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(13,51),ATTRB=(ASKIP),LENGTH=1
INTEX    DFHMDF   POS=(13,57),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(13,67),ATTRB=(ASKIP),LENGTH=1
PERIOD3  DFHMDF   POS=(13,71),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,            *
               PICOUT='ZZ9'
         DFHMDF   POS=(13,75),ATTRB=(ASKIP),LENGTH=1
ODEBT    DFHMDI   SIZE=(6,80),LINE=14,COLUMN=1
         DFHMDF   POS=(01,4),ATTRB=(PROT,BRT),LENGTH=12,                  *
               INITIAL='**OLD DEBT**'
         DFHMDF   POS=(02,9),ATTRB=(PROT,NORM),LENGTH=9,                  *
               INITIAL='LOAN TYPE'
         DFHMDF   POS=(02,26),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='PRINCIPAL'
         DFHMDF   POS=(02,43),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='DEBT REMAIN'
         DFHMDF   POS=(02,61),ATTRB=(PROT,NORM),LENGTH=7,                 *
               INITIAL='PERIOD#'
OLDL1    DFHMDF   POS=(03,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(03,14),ATTRB=(ASKIP),LENGTH=1
PRINCE1  DFHMDF   POS=(03,28),ATTRB=(UNPROT,BRT),LENGTH=5,                *
               PICOUT='Z,ZZ9'
         DFHMDF   POS=(03,34),ATTRB=(ASKIP),LENGTH=1
DEBTRE1  DFHMDF   POS=(03,43),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(03,51),ATTRB=(ASKIP),LENGTH=1
FPERID1  DFHMDF   POS=(03,61),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,            *
               PICOUT='ZZ9'
         DFHMDF   POS=(03,65),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF   POS=(03,67),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(03,79),ATTRB=(ASKIP),LENGTH=1
OLDL2    DFHMDF   POS=(04,12),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(04,14),ATTRB=(ASKIP),LENGTH=1
PRINCE2  DFHMDF   POS=(04,26),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(04,34),ATTRB=(ASKIP),LENGTH=1
DEBTRE2  DFHMDF   POS=(04,43),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(04,51),ATTRB=(ASKIP),LENGTH=1
FPERID2  DFHMDF   POS=(04,61),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,            *
               PICOUT='ZZ9'
         DFHMDF   POS=(04,65),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF   POS=(04,67),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(04,79),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF   POS=(05,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF   POS=(05,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(06,19),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='PREVIOUS DEBT  '
KDEBT    DFHMDF   POS=(06,36),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(06,38),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(06,41),ATTRB=(PROT,NORM),LENGTH=16,                *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD   TYPE=FINAL
         END
```

```
M=MAPPLY1
         PRINT NOGEN
APPY     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPPLY1  DFHMDI   SIZE=(24,80),LINE=20,COLUMN=1
         DFHMDF   POS=(01,4),ATTRB=(PROT,BRT),LENGTH=9,                   *
               INITIAL='**APPLY**'
         DFHMDF   POS=(01,19),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LOAN TYPE'
LOTYPEA  DFHMDF   POS=(01,30),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=1,         *
               INITIAL='-'
         DFHMDF   POS=(01,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(01,34),ATTRB=(PROT,NORM),LENGTH=37,                *
               INITIAL='(1 = EMER,2 = ORDI,3 = MORT,4 = LAND,'
         DFHMDF   POS=(01,72),ATTRB=(PROT,BRT),LENGTH=6,                  *
               INITIAL='0= END'
         DFHMDF   POS=(01,79),ATTRB=(PROT,NORM),LENGTH=1,                 *
               INITIAL=')'
LOAPPLY  DFHMDF   POS=(02,3),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,             *
               JUSTIFY=(RIGHT,ZERO),INITIAL='------'
         DFHMDF   POS=(02,10),ATTRB=(ASKIP),LENGTH=1
PERIODL  DFHMDF   POS=(02,16),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,             *
               JUSTIFY=(RIGHT,ZERO),INITIAL='---'
         DFHMDF   POS=(02,20),ATTRB=(ASKIP),LENGTH=1
APERIOD  DFHMDF   POS=(02,28),ATTRB=(UNPROT,BRT),LENGTH=5,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(02,35),ATTRB=(ASKIP),LENGTH=1
LPERIOD  DFHMDF   POS=(02,44),ATTRB=(UNPROT,BRT),LENGTH=5,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(02,51),ATTRB=(ASKIP),LENGTH=1
LINTA    DFHMDF   POS=(02,58),ATTRB=(UNPROT,BRT),LENGTH=8,                *
               PICOUT='Z,ZZ9.99'
         DFHMDF   POS=(02,67),ATTRB=(ASKIP),LENGTH=1
TOTALM   DFHMDF   POS=(02,70),ATTRB=(UNPROT,BRT),LENGTH=9,                *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(02,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(03,2),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='(PRINCIPAL)'
         DFHMDF   POS=(03,15),ATTRB=(PROT,NORM),LENGTH=8,                 *
               INITIAL='(PERIOD)'
         DFHMDF   POS=(03,25),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='(MONTHLY PRINC)'
         DFHMDF   POS=(03,42),ATTRB=(PROT,NORM),LENGTH=13,                *
               INITIAL='(FIRST PRINC)'
         DFHMDF   POS=(03,57),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='(INTEREST)'
         DFHMDF   POS=(03,69),ATTRB=(PROT,NORM),LENGTH=12,                *
               INITIAL='(FIRST PAID)'
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.MAPPLY1

         PRINT NOGEN
APPY     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPPLY1  DFHMDI   SIZE=(3,80),LINE=20,COLUMN=1
         DFHMDF   POS=(01,4),ATTRB=(PROT,BRT),LENGTH=9,                 *
               INITIAL='**APPLY**'
         DFHMDF   POS=(01,19),ATTRB=(PROT,NORM),LENGTH=9,               *
               INITIAL='LOAN TYPE'
LOTYPEA  DFHMDF   POS=(01,30),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=1,       *
               INITIAL='-'
         DFHMDF   POS=(01,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(01,34),ATTRB=(PROT,NORM),LENGTH=37,              *
               INITIAL='(1 = EMER,2 = ORDI,3 = MORT,4 = LAND,'
         DFHMDF   POS=(01,72),ATTRB=(PROT,BRT),LENGTH=6,                *
               INITIAL='9= END'
         DFHMDF   POS=(01,79),ATTRB=(PROT,NORM),LENGTH=1,               *
               INITIAL=')'
LOAPPLY  DFHMDF   POS=(02,3),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,           *
               JUSTIFY=(RIGHT,ZERO),INITIAL='------'
         DFHMDF   POS=(02,10),ATTRB=(ASKIP),LENGTH=1
PERIODL  DFHMDF   POS=(02,16),ATTRB=(UNPROT,BRT,NUM),LENGTH=3,          *
               JUSTIFY=(RIGHT,ZERO),INITIAL='---'
         DFHMDF   POS=(02,20),ATTRB=(ASKIP),LENGTH=1
APERID1  DFHMDF   POS=(02,28),ATTRB=(UNPROT,BRT),LENGTH=5,              *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(02,35),ATTRB=(ASKIP),LENGTH=1
LPERID   DFHMDF   POS=(02,44),ATTRB=(UNPROT,BRT),LENGTH=5,              *
               PICOUT='ZZ,ZZ9'
         DFHMDF   POS=(02,51),ATTRB=(ASKIP),LENGTH=1
LINTA    DFHMDF   POS=(02,58),ATTRB=(UNPROT,BRT),LENGTH=8,              *
               PICOUT='Z,ZZ9.99'
         DFHMDF   POS=(02,67),ATTRB=(ASKIP),LENGTH=1
TOTALM   DFHMDF   POS=(02,70),ATTRB=(UNPROT,BRT),LENGTH=9,              *
               PICOUT='ZZ,ZZ9.99'
         DFHMDF   POS=(02,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(03,2),ATTRB=(PROT,NORM),LENGTH=11,               *
               INITIAL='(PRINCIPAL)'
         DFHMDF   POS=(03,15),ATTRB=(PROT,NORM),LENGTH=8,               *
               INITIAL='(PERIOD)'
         DFHMDF   POS=(03,25),ATTRB=(PROT,NORM),LENGTH=15,              *
               INITIAL='(MONTHLY PRINC)'
         DFHMDF   POS=(03,42),ATTRB=(PROT,NORM),LENGTH=13,              *
               INITIAL='(FIRST PRINC)'
         DFHMDF   POS=(03,57),ATTRB=(PROT,NORM),LENGTH=10,              *
               INITIAL='(INTEREST)'
         DFHMDF   POS=(03,68),ATTRB=(PROT,NORM),LENGTH=12,              *
               INITIAL='(FIRST PAID)'
         DFHMSD   TYPE=FINAL
         END
```

```
*=MWRITE
         PRINT NOGEN
CWRITE   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MWRITE   DFHMDI   SIZE=(1,80),LINE=23,COLUMN=1
         DFHMDF   POS=(01,4),ATTRB=(PROT,NORM),LENGTH=7,                 *
               INITIAL='***O.K ?'
OPTION2  DFHMDF   POS=(01,13),ATTRB=(UNPROT,BRT,IC),LENGTH=1,            *
               INITIAL='-'
         DFHMDF   POS=(01,15),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(01,18),ATTRB=(PROT,NORM),LENGTH=15,               *
               INITIAL='(Y = YES,N = NO)'
         DFHMDF   POS=(01,41),ATTRB=(PROT,NORM),LENGTH=8,                *
               INITIAL='APPLY# ?'
MOAPY    DFHMDF   POS=(01,50),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,           *
               INITIAL='----'
         DFHMDF   POS=(01,55),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.MWRITE

         PRINT NOGEN
CMPITE   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MWRITE   DFHMDI   SIZE=(1,80),LINE=23,COLUMN=1
         DFHMDF   POS=(01,4),ATTRB=(PROT,NORM),LENGTH=7,              *
               INITIAL='**O.K ?'
OPTION2  DFHMDF   POS=(01,13),ATTRB=(UNPROT,BRT,IC),LENGTH=1,        *
               INITIAL='-'
         DFHMDF   POS=(01,15),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(01,18),ATTRB=(PROT,NORM),LENGTH=15,           *
               INITIAL='(Y = YES,N = NO)'
         DFHMDF   POS=(01,41),ATTRB=(PROT,NORM),LENGTH=8,            *
               INITIAL='APPLY# ?'
NOAPY    DFHMDF   POS=(01,50),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,       *
               INITIAL='----'
         DFHMDF   POS=(01,55),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

M=MAPERRL

```
         PRINT NOGEN
CMAP     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1  DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
ERR      DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,             *
               JUSTIFY=(LEFT,BLANK)
         DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.MAPEPR1

          PRINT NOGEN
CMAP      DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPEPR1   DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
LERR      DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,              *
               JUSTIFY=(LEFT,BLANK)
          DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
          DFHMSD   TYPE=FINAL
          END
```

ค.๑.๔ แม็บและดีเซ็คที่ใช้ในโปรแกรม P04

```
M=INPUTM
         PRINT NOGEN
GUAM     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=FREEKB,DATA=FIELD,TIOAPFX=YES
INPUTM   DFHMDF   SIZE=(5,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,               *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
MDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=13,              *
               INITIAL='GUARANTEE MAN'
         DFHMDF   POS=(4,6),ATTRB=(PROT,NORM),LENGTH=14,               *
               INITIAL='ID-GUARANTOR 2'
KEYID2   DFHMDF   POS=(4,21),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=6,       *
               INITIAL='------'
         DFHMDF   POS=(4,28),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,31),ATTRB=(PROT,NORM),LENGTH=4,INITIAL='NAME'
NAME2    DFHMDF   POS=(4,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(4,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.INPUTM
         PRINT NOGEN
GUAM     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=FREEKB,DATA=FIELD,TIOAPFX=YES
INPUTM   DFHMDI   SIZE=(5,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,                *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
MDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=13,               *
               INITIAL='GUARANTEE MAN'
         DFHMDF   POS=(4,6),ATTRB=(PROT,NORM),LENGTH=14,                *
               INITIAL='ID-GUARANTOR ?'
KEYIDG   DFHMDF   POS=(4,21),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=6,        *
               INITIAL='------'
         DFHMDF   POS=(4,28),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,31),ATTRB=(PROT,NORM),LENGTH=4,INITIAL='NAME'
NAMEG    DFHMDF   POS=(4,38),ATTRB=(UNPROT,BRT),LENGTH=40
         DFHMDF   POS=(4,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
M=DPGUA
         PRINT NOGEN
GUAMAN   DFHMSD  TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,
               CTRL=(FREEKB,DATA=FIELD),TIOAPFX=YES
DPGUA    DFHMDI  SIZE=(15,80),LINE=6,COLUMN=1
         DFHMDF  POS=(1,6),ATTRB=(PROT,NORM),LENGTH=7,
               INITIAL='MARITAL'
MARRY    DFHMDF  POS=(1,15),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF  POS=(1,17),ATTRB=(ASKIP),LENGTH=1
MARIT    DFHMDF  POS=(1,19),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(1,31),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(1,41),ATTRB=(PROT,NORM),LENGTH=9,
               INITIAL='ID-SPOUSE'
IDSPOU   DFHMDF  POS=(1,60),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(1,67),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(2,06),ATTRB=(PROT,NORM),LENGTH=15,
               INITIAL='AMOUNT-GUARANTY'
AMTI     DFHMDF  POS=(2,25),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(2,33),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(3,6),ATTRB=(PROT,NORM),LENGTH=14,
               INITIAL='GUA-LIMIT-USED'
USEL     DFHMDF  POS=(3,25),ATTRB=(UNPROT,BRT),LENGTH=6,
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(3,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(3,41),ATTRB=(PROT,NORM),LENGTH=14,
               INITIAL='NET-GUA-REMAIN'
NREM     DFHMDF  POS=(3,60),ATTRB=(UNPROT,BRT),LENGTH=6,
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(3,67),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(05,6),ATTRB=(PROT,NORM),LENGTH=12,
               INITIAL='ID-GUARANTED'
         DFHMDF  POS=(05,34),ATTRB=(PROT,NORM),LENGTH=15,
               INITIAL='GUARANTEE LIMIT'
         DFHMDF  POS=(05,62),ATTRB=(PROT,NORM),LENGTH=14,
               INITIAL='AMT-GUA-REMAIN'
IDGUAR1  DFHMDF  POS=(07,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(07,15),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF  POS=(07,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(07,29),ATTRB=(ASKIP),LENGTH=1
GUALIM1  DFHMDF  POS=(07,38),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(07,46),ATTRB=(ASKIP),LENGTH=1
GUAREM1  DFHMDF  POS=(07,65),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(07,73),ATTRB=(ASKIP),LENGTH=1
IDGUAR2  DFHMDF  POS=(08,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(08,15),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF  POS=(08,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(08,29),ATTRB=(ASKIP),LENGTH=1
GUALIM2  DFHMDF  POS=(08,38),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(08,46),ATTRB=(ASKIP),LENGTH=1
GUAREM2  DFHMDF  POS=(08,65),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(08,73),ATTRB=(ASKIP),LENGTH=1
IDGUAR3  DFHMDF  POS=(09,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(09,15),ATTRB=(ASKIP),LENGTH=1
TORP3    DFHMDF  POS=(09,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(09,29),ATTRB=(ASKIP),LENGTH=1
GUALIM3  DFHMDF  POS=(09,38),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(09,46),ATTRB=(ASKIP),LENGTH=1
GUAREM3  DFHMDF  POS=(09,65),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(09,73),ATTRB=(ASKIP),LENGTH=1
IDGUAR4  DFHMDF  POS=(10,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(10,15),ATTRB=(ASKIP),LENGTH=1
TORP4    DFHMDF  POS=(10,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(10,29),ATTRB=(ASKIP),LENGTH=1
GUALIM4  DFHMDF  POS=(10,38),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(10,46),ATTRB=(ASKIP),LENGTH=1
GUAREM4  DFHMDF  POS=(10,65),ATTRB=(UNPROT,BRT),LENGTH=7,
               PICOUT='ZZZ,ZZ9'
```

```
         DFHMDF   POS=(10,73),ATTRB=(ASKIP),LENGTH=1
IDGUAE5  DFHMDF   POS=(11,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF   POS=(11,15),ATTRB=(ASKIP),LENGTH=1
TORP5    DFHMDF   POS=(11,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(11,29),ATTRB=(ASKIP),LENGTH=1
QUALI5   DFHMDF   POS=(11,38),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ.ZZ9'
         DFHMDF   POS=(11,46),ATTRB=(ASKIP),LENGTH=1
QUARE5   DFHMDF   POS=(11,65),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ.ZZ9'
         DFHMDF   POS=(11,73),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF   POS=(13,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF   POS=(13,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(14,6),ATTRB=(PROT,NORM),LENGTH=17,                 *
               INITIAL='PREVIOUS STATUS ?'
PREVS    DFHMDF   POS=(14,30),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(14,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(14,46),ATTRB=(PROT,NORM),LENGTH=17,                *
               INITIAL='(Y = YES, N = NO)'
         DFHMSD   TYPE=FINAL
                  END
```

```
D=C.DPGUA
         PRINT NOGEN
GUAMAN   DFHMSD  TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,DATA=FIELD,TIOAPFX=YES
DPGUA    DFHMDI  SIZE=(15,80),LINE=6,COLUMN=1
         DFHMDF  POS=(1,6),ATTRB=(PROT,NORM),LENGTH=7,                  *
               INITIAL='MARITAL'
MARRY    DFHMDF  POS=(1,15),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF  POS=(1,17),ATTRB=(ASKIP),LENGTH=1
MARIT    DFHMDF  POS=(1,19),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(1,31),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(1,41),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='ID-SPOUSE'
IDSPOU   DFHMDF  POS=(1,60),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(1,67),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(2,06),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='AMOUNT-GUARANTY'
AMTL     DFHMDF  POS=(2,25),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(2,33),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(3,6),ATTRB=(PROT,NORM),LENGTH=14,                 *
               INITIAL='GUA-LIMIT-USED'
USEL     DFHMDF  POS=(3,25),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(3,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(3,41),ATTRB=(PROT,NORM),LENGTH=14,                *
               INITIAL='NET-GUA-REMAIN'
NREM     DFHMDF  POS=(3,60),ATTRB=(UNPROT,BRT),LENGTH=6,                *
               PICOUT='ZZ,ZZ9'
         DFHMDF  POS=(3,67),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(05,6),ATTRB=(PROT,NORM),LENGTH=12,                *
               INITIAL='ID-GUARANTED'
         DFHMDF  POS=(05,34),ATTRB=(PROT,NORM),LENGTH=15,               *
               INITIAL='GUARANTEE LIMIT'
         DFHMDF  POS=(05,62),ATTRB=(PROT,NORM),LENGTH=14,               *
               INITIAL='AMT-GUA-REMAIN'
IDGUAE1  DFHMDF  POS=(07,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(07,15),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF  POS=(07,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(07,29),ATTRB=(ASKIP),LENGTH=1
GUALIM1  DFHMDF  POS=(07,38),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(07,46),ATTRB=(ASKIP),LENGTH=1
GUAREM1  DFHMDF  POS=(07,65),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(07,73),ATTRB=(ASKIP),LENGTH=1
IDGUAE2  DFHMDF  POS=(08,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(08,15),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF  POS=(08,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(08,29),ATTRB=(ASKIP),LENGTH=1
GUALIM2  DFHMDF  POS=(08,38),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(08,46),ATTRB=(ASKIP),LENGTH=1
GUAREM2  DFHMDF  POS=(09,65),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(08,73),ATTRB=(ASKIP),LENGTH=1
IDGUAE3  DFHMDF  POS=(09,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(09,15),ATTRB=(ASKIP),LENGTH=1
TORP3    DFHMDF  POS=(09,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(09,29),ATTRB=(ASKIP),LENGTH=1
GUALIM3  DFHMDF  POS=(09,38),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(09,46),ATTRB=(ASKIP),LENGTH=1
GUAREM3  DFHMDF  POS=(09,65),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(09,73),ATTRB=(ASKIP),LENGTH=1
IDGUAE4  DFHMDF  POS=(10,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF  POS=(10,15),ATTRB=(ASKIP),LENGTH=1
TORP4    DFHMDF  POS=(10,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF  POS=(10,29),ATTRB=(ASKIP),LENGTH=1
GUALIM4  DFHMDF  POS=(10,38),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF  POS=(10,46),ATTRB=(ASKIP),LENGTH=1
GUAREM4  DFHMDF  POS=(10,65),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
```

```
         DFHMDF    POS=(10,73),ATTRB=(ASKIP),LENGTH=1
IDGUAE5  DFHMDF    POS=(11,8),ATTRB=(UNPROT,BRT,NUM),LENGTH=6
         DFHMDF    POS=(11,15),ATTRB=(ASKIP),LENGTH=1
TORP5    DFHMDF    POS=(11,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF    POS=(11,29),ATTRB=(ASKIP),LENGTH=1
GUALIM5  DFHMDF    POS=(11,38),ATTRB=(UNPROT,BRT),LENGTH=7,          *
               PICOUT='ZZZ,ZZ9'
         DFHMDF    POS=(11,46),ATTRB=(ASKIP),LENGTH=1
GUAREM5  DFHMDF    POS=(11,65),ATTRB=(UNPROT,BRT),LENGTH=7,          *
               PICOUT='ZZZ,ZZ9'
         DFHMDF    POS=(11,73),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF    POS=(13,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF    POS=(13,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF    POS=(14,6),ATTRB=(PROT,NORM),LENGTH=17,           *
               INITIAL='PREVIOUS STATUS ?'
PPEVS    DFHMDF    POS=(14,30),ATTRB=(UNPROT,BRT,IC),LENGTH=1,       *
               INITIAL='-'
         DFHMDF    POS=(14,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF    POS=(14,46),ATTRB=(PROT,NORM),LENGTH=17,          *
               INITIAL='(Y = YES, N = NO)'
         DFHMSD    TYPE=FINAL
                   END
```

```
M=AMAN

         PRINT NOGEN
AMAN1    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
AMAN     DFHMDI   SIZE=(3,80),LINE=21,COLUMN=1
         DFHMDF   POS=(1,6),ATTRB=(PROT,NORM),LENGTH=23,                 *
               INITIAL='ACCEPT AS A GUARANTOR ?'
YORN     DFHMDF   POS=(1,30),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(1,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,36),ATTRB=(PROT,NORM),LENGTH=29,                *
               INITIAL='(Y = YES, N = NO)    APPLY# ?'
NOAP     DFHMDF   POS=(1,67),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               INITIAL='----'
         DFHMDF   POS=(1,72),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,6),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER ?'
KEYID    DFHMDF   POS=(2,30),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,            *
               INITIAL='------'
         DFHMDF   POS=(2,37),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,49),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='AMOUNT TO USE ?'
AMTW     DFHMDF   POS=(2,66),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(2,72),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,6),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='USE SHARE ?'
USHARE   DFHMDF   POS=(3,31),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(3,37),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,50),ATTRB=(PROT,NORM),LENGTH=13,                *
               INITIAL='USE DEPOSIT ?'
USDEPO   DFHMDF   POS=(3,66),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(3,72),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
D=C.AMAN

         PRINT NOGEN
AMAN1    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
AMAN     DFHMDI   SIZE=(3,80),LINE=21,COLUMN=1
         DFHMDF   POS=(1,6),ATTRB=(PROT,NORM),LENGTH=23,                 *
               INITIAL='ACCEPT AS A GUARANTOR ?'
YORN     DFHMDF   POS=(1,30),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(1,32),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,36),ATTRB=(PROT,NORM),LENGTH=28,                *
               INITIAL='(Y = YES, N = NO)   APPLY# ?'
NOAP     DFHMDF   POS=(1,67),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               INITIAL='----'
         DFHMDF   POS=(1,72),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,6),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='ID-MEMBER ?'
KEYID    DFHMDF   POS=(2,30),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,            *
               INITIAL='------'
         DFHMDF   POS=(2,37),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,49),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='AMOUNT TO USE ?'
AMTW     DFHMDF   POS=(2,66),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(2,72),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,6),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='USE SHARE ?'
USHARE   DFHMDF   POS=(3,31),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(3,37),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,50),ATTRB=(PROT,NORM),LENGTH=13,                *
               INITIAL='USE DEPOSIT ?'
USDEPO   DFHMDF   POS=(3,66),ATTRB=(UNPROT,BRT,NUM),LENGTH=5,            *
               INITIAL='-----'
         DFHMDF   POS=(3,72),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
M=MAPERR1

         PRINT NOGEN
CMAP     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1  DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
ERR      DFHMDF   POS=(01,11),ATTRB=(UNPROT,BRT),LENGTH=69,            *
               JUSTIFY=(LEFT,BLANK)
         DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.MAPERR1

         PRINT NOGEN
CMAP     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1  DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
LERR     DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,             *
               JUSTIFY=(LEFT,BLANK)
         DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

ค.๑.๕ แม็บและดีเซ็คที่ใช้ในโปรแกรม P05

```
M=INPUTL
         PRINT NOGEN
GUAL     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=FREEKB,DATA=FIELD,TIOAPFX=YES
INPUTL   DFHMDI   SIZE=(24,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,               *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
KDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,              *
               INITIAL='LAND DESCRIPTION'
         DFHMDF   POS=(4,26),ATTRB=(PROT,NORM),LENGTH=18,              *
               INITIAL='GUARANTEE LAND # ?'
LANDID   DFHMDF   POS=(4,46),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=9,       *
               INITIAL='---------'
         DFHMDF   POS=(4,56),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.INPUTL
         PRINT NOGEN
QUAL     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
               CTRL=FREEKB,DATA=FIELD,TIOAPFX=YES
INPUTI   DFHMDI   SIZE=(24,80),LINE=1,COLUMN=1
         DFHMDF   POS=(1,18),ATTRB=(PROT,BRT),LENGTH=46,              *
               INITIAL='SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY'
KDATE    DFHMDF   POS=(1,68),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(1,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,33),ATTRB=(PROT,NORM),LENGTH=16,             *
               INITIAL='LAND DESCRIPTION'
         DFHMDF   POS=(4,26),ATTRB=(PROT,NORM),LENGTH=18,             *
               INITIAL='GUARANTEE LAND # 7'
LANDID   DFHMDF   POS=(4,46),ATTRB=(UNPROT,BRT,NUM,IC),LENGTH=9,      *
               INITIAL='---------'
         DFHMDF   POS=(4,56),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

```
M=OLANDES

         PRINT NOGEN
LANDES   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
OLANDES  DFHMDI   SIZE=(16,80),LINE=5,COLUMN=1
         DFHMDF   POS=(1,10),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LOAN TYPE'
LNTYPE   DFHMDF   POS=(1,20),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(1,22),ATTRB=(ASKIP),LENGTH=1
SEXTPA   DFHMDF   POS=(1,24),ATTRB=(UNPROT,NORM),LENGTH=1A
         DFHMDF   POS=(1,43),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,47),ATTRB=(PROT,NORM),LENGTH=12,                *
               INITIAL='MEN PER LAND'
MENNUM   DFHMDF   POS=(1,64),ATTRB=(UNPROT,BRT,NUM),LENGTH=1
         DFHMDF   POS=(1,66),ATTRB=(ASKIP),LENGTH=1
LBLANK   DFHMDF   POS=(2,2),ATTRB=(UNPROT,NORM),LENGTH=7A
         DFHMDF   POS=(2,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,10),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LAND-AREA'
         DFHMDF   POS=(3,22),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='EVAL/UNIT'
         DFHMDF   POS=(3,34),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='EVAL-PRICE'
         DFHMDF   POS=(3,47),ATTRB=(PROT,NORM),LENGTH=13,                *
               INITIAL='EVAL-BUILDING'
         DFHMDF   POS=(3,63),ATTRB=(PROT,NORM),LENGTH=14,                *
               INITIAL='FIRE INSURANCE'
LANDAR   DFHMDF   POS=(4,11),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               PICOUT='ZZZ9'
         DFHMDF   POS=(4,16),ATTRB=(ASKIP),LENGTH=1
EVAPUT   DFHMDF   POS=(4,22),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(4,30),ATTRB=(ASKIP),LENGTH=1
EVAPPI   DFHMDF   POS=(4,34),ATTRB=(UNPROT,BRT),LENGTH=9,                *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(4,44),ATTRB=(ASKIP),LENGTH=1
EVABUD   DFHMDF   POS=(4,48),ATTRB=(UNPROT,BRT),LENGTH=9,                *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(4,58),ATTRB=(ASKIP),LENGTH=1
FIREIN   DFHMDF   POS=(4,66),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(4,74),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(06,10),ATTRB=(PROT,NORM),LENGTH=13,               *
               INITIAL='EVALUATE-DATE'
EVADAT   DFHMDF   POS=(06,47),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(06,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(07,10),ATTRB=(PROT,NORM),LENGTH=27,               *
               INITIAL='AMOUNT GUARANTEE SECURITIES'
AMTGUA   DFHMDF   POS=(07,50),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(07,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(08,10),ATTRB=(UNPROT,NORM),LENGTH=17,             *
               INITIAL='REMAIN SECURITIES'
REMSEC   DFHMDF   POS=(08,50),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,Z79'
         DFHMDF   POS=(08,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(09,06),ATTRB=(PROT,NORM),LENGTH=12,               *
               INITIAL='ID-GUARANTED'
         DFHMDF   POS=(09,31),ATTRB=(PROT,NORM),LENGTH=9,                *
               INITIAL='PRINCIPAL'
         DFHMDF   POS=(09,47),ATTRB=(PROT,NORM),LENGTH=10,               *
               INITIAL='SECUR-USED'
         DFHMDF   POS=(09,63),ATTRB=(PROT,NORM),LENGTH=11,               *
               INITIAL='OTHER SECUR'
GUAED1   DFHMDF   POS=(10,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(10,15),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF   POS=(10,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(10,29),ATTRB=(ASKIP),LENGTH=1
PRINC1   DFHMDF   POS=(10,31),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,39),ATTRB=(ASKIP),LENGTH=1
SECU1    DFHMDF   POS=(10,48),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,56),ATTRB=(ASKIP),LENGTH=1
```

```
OTHER1   DFHMDF   POS=(10,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,71),ATTRB=(ASKIP),LENGTH=1
GUAED2   DFHMDF   POS=(11,08),ATTRB=(UNPROT,BRT),LENGTH=5
         DFHMDF   POS=(11,15),ATTRB=(ASKIP),LENGTH=1
TOPP2    DFHMDF   POS=(11,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(11,29),ATTRB=(ASKIP),LENGTH=1
PRINC2   DFHMDF   POS=(11,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,39),ATTRB=(ASKIP),LENGTH=1
SECU2    DFHMDF   POS=(11,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,56),ATTRB=(ASKIP),LENGTH=1
OTHER2   DFHMDF   POS=(11,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,71),ATTRB=(ASKIP),LENGTH=1
GUAED3   DFHMDF   POS=(12,08),ATTRB=(UNPROT,BRT),LENGTH=5
         DFHMDF   POS=(12,15),ATTRB=(ASKIP),LENGTH=1
TOPP3    DFHMDF   POS=(12,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(12,29),ATTRB=(ASKIP),LENGTH=1
PRINC3   DFHMDF   POS=(12,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,39),ATTRB=(ASKIP),LENGTH=1
SECU3    DFHMDF   POS=(12,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,56),ATTRB=(ASKIP),LENGTH=1
OTHER3   DFHMDF   POS=(12,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,71),ATTRB=(ASKIP),LENGTH=1
GUAED4   DFHMDF   POS=(13,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(13,15),ATTRB=(ASKIP),LENGTH=1
TOPP4    DFHMDF   POS=(13,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(13,29),ATTRB=(ASKIP),LENGTH=1
PRINC4   DFHMDF   POS=(13,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,39),ATTRB=(ASKIP),LENGTH=1
SECU4    DFHMDF   POS=(13,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,56),ATTRB=(ASKIP),LENGTH=1
OTHER4   DFHMDF   POS=(13,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,71),ATTRB=(ASKIP),LENGTH=1
GUAED5   DFHMDF   POS=(14,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(14,15),ATTRB=(ASKIP),LENGTH=1
TOPP5    DFHMDF   POS=(14,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(14,29),ATTRB=(ASKIP),LENGTH=1
PRINC5   DFHMDF   POS=(14,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,39),ATTRB=(ASKIP),LENGTH=1
SECU5    DFHMDF   POS=(14,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,56),ATTRB=(ASKIP),LENGTH=1
OTHER5   DFHMDF   POS=(14,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,71),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF   POS=(15,02),ATTRB=(UNPROT,BRT),LENGTH=77
         DFHMDF   POS=(15,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(16,20),ATTRB=(PROT,NORM),LENGTH=17,                *
               INITIAL='PREVIOUS STATUS ?'
CSTAT    DFHMDF   POS=(16,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(16,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(16,43),ATTRB=(UNPROT,NORM),LENGTH=16,              *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD   TYPE=FINAL
         END
```

```
D=C.CLANDES

         PRINT NOGEN
LANDES   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
CLANDES  DFHMDI   SIZE=(16,80),LINE=5,COLUMN=1
         DFHMDF   POS=(1,10),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LOAN TYPE'
LNTYPE   DFHMDF   POS=(1,20),ATTRB=(UNPROT,BRT),LENGTH=1
         DFHMDF   POS=(1,22),ATTRB=(ASKIP),LENGTH=1
SEXTPA   DFHMDF   POS=(1,24),ATTRB=(UNPROT,NORM),LENGTH=18
         DFHMDF   POS=(1,43),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,47),ATTRB=(PROT,NORM),LENGTH=12,                *
               INITIAL='MEN PER LAND'
MENNUM   DFHMDF   POS=(1,64),ATTRB=(UNPROT,BRT,NUM),LENGTH=1
         DFHMDF   POS=(1,66),ATTRB=(ASKIP),LENGTH=1
LBLANK   DFHMDF   POS=(2,2),ATTRB=(UNPROT,NORM),LENGTH=76
         DFHMDF   POS=(2,79),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,10),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='LAND-AREA'
         DFHMDF   POS=(3,22),ATTRB=(PROT,NORM),LENGTH=9,                 *
               INITIAL='EVAL/UNIT'
         DFHMDF   POS=(3,34),ATTRB=(PROT,NORM),LENGTH=10,                *
               INITIAL='EVAL-PRICE'
         DFHMDF   POS=(3,47),ATTRB=(PROT,NORM),LENGTH=13,                *
               INITIAL='EVAL-BUILDING'
         DFHMDF   POS=(3,63),ATTRB=(PROT,NORM),LENGTH=14,                *
               INITIAL='FIRE INSURANCE'
LANDAR   DFHMDF   POS=(4,11),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               PICOUT='ZZZ9'
         DFHMDF   POS=(4,16),ATTRB=(ASKIP),LENGTH=1
EVAPUT   DFHMDF   POS=(4,22),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(4,30),ATTRB=(ASKIP),LENGTH=1
EVAPRI   DFHMDF   POS=(4,34),ATTRB=(UNPROT,BRT),LENGTH=9,                *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(4,44),ATTRB=(ASKIP),LENGTH=1
EVABUD   DFHMDF   POS=(4,48),ATTRB=(UNPROT,BRT),LENGTH=9,                *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(4,58),ATTRB=(ASKIP),LENGTH=1
FIREIN   DFHMDF   POS=(4,66),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(4,74),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(06,10),ATTRB=(PROT,NORM),LENGTH=13,               *
               INITIAL='EVALUATE-DATE'
EVADAT   DFHMDF   POS=(06,47),ATTRB=(UNPROT,BRT),LENGTH=10
         DFHMDF   POS=(06,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(07,10),ATTRB=(PROT,NORM),LENGTH=27,               *
               INITIAL='AMOUNT GUARANTEE SECURITIES'
AMTGUA   DFHMDF   POS=(07,50),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,999'
         DFHMDF   POS=(07,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(08,10),ATTRB=(UNPROT,NORM),LENGTH=17,             *
               INITIAL='REMAIN SECURITIES'
REMSEC   DFHMDF   POS=(08,50),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(08,58),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(09,06),ATTRB=(PROT,NORM),LENGTH=12,               *
               INITIAL='ID-GUARANTED'
         DFHMDF   POS=(09,31),ATTRB=(PROT,NORM),LENGTH=9,                *
               INITIAL='PRINCIPAL'
         DFHMDF   POS=(09,47),ATTRB=(PROT,NORM),LENGTH=10,               *
               INITIAL='SECUR-USED'
         DFHMDF   POS=(09,63),ATTRB=(PROT,NORM),LENGTH=11,               *
               INITIAL='OTHER SECUR'
GUAED1   DFHMDF   POS=(10,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(10,15),ATTRB=(ASKIP),LENGTH=1
TORP1    DFHMDF   POS=(10,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(10,29),ATTRB=(ASKIP),LENGTH=1
PRINC1   DFHMDF   POS=(10,31),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,39),ATTRB=(ASKIP),LENGTH=1
SECU1    DFHMDF   POS=(10,48),ATTRB=(UNPROT,BRT),LENGTH=7,               *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,56),ATTRB=(ASKIP),LENGTH=1
```

```
OTHER1   DFHMDF   POS=(10,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(10,71),ATTRB=(ASKIP),LENGTH=1
GUAED2   DFHMDF   POS=(11,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(11,15),ATTRB=(ASKIP),LENGTH=1
TORP2    DFHMDF   POS=(11,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(11,29),ATTRB=(ASKIP),LENGTH=1
PRINC2   DFHMDF   POS=(11,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,39),ATTRB=(ASKIP),LENGTH=1
SECU2    DFHMDF   POS=(11,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,56),ATTRB=(ASKIP),LENGTH=1
OTHER2   DFHMDF   POS=(11,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(11,71),ATTRB=(ASKIP),LENGTH=1
GUAED3   DFHMDF   POS=(12,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(12,15),ATTRB=(ASKIP),LENGTH=1
TORP3    DFHMDF   POS=(12,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(12,29),ATTRB=(ASKIP),LENGTH=1
PRINC3   DFHMDF   POS=(12,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,39),ATTRB=(ASKIP),LENGTH=1
SECU3    DFHMDF   POS=(12,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,56),ATTRB=(ASKIP),LENGTH=1
OTHER3   DFHMDF   POS=(12,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(12,71),ATTRB=(ASKIP),LENGTH=1
GUAED4   DFHMDF   POS=(13,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(13,15),ATTRB=(ASKIP),LENGTH=1
TORP4    DFHMDF   POS=(13,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(13,29),ATTRB=(ASKIP),LENGTH=1
PRINC4   DFHMDF   POS=(13,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,39),ATTRB=(ASKIP),LENGTH=1
SECU4    DFHMDF   POS=(13,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,56),ATTRB=(ASKIP),LENGTH=1
OTHER4   DFHMDF   POS=(13,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(13,71),ATTRB=(ASKIP),LENGTH=1
GUAED5   DFHMDF   POS=(14,08),ATTRB=(UNPROT,BRT),LENGTH=6
         DFHMDF   POS=(14,15),ATTRB=(ASKIP),LENGTH=1
TORP5    DFHMDF   POS=(14,17),ATTRB=(UNPROT,NORM),LENGTH=11
         DFHMDF   POS=(14,29),ATTRB=(ASKIP),LENGTH=1
PRINC5   DFHMDF   POS=(14,31),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,39),ATTRB=(ASKIP),LENGTH=1
SECU5    DFHMDF   POS=(14,48),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,56),ATTRB=(ASKIP),LENGTH=1
OTHER5   DFHMDF   POS=(14,63),ATTRB=(UNPROT,BRT),LENGTH=7,                *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(14,71),ATTRB=(ASKIP),LENGTH=1
LINE1    DFHMDF   POS=(15,02),ATTRB=(UNPROT,BRT),LENGTH=77
         DFHMDF   POS=(15,80),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(16,20),ATTRB=(PROT,NORM),LENGTH=17,                *
               INITIAL='PREVIOUS STATUS ?'
CSTAT    DFHMDF   POS=(16,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(16,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(16,43),ATTRB=(UNPROT,NORM),LENGTH=16,              *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD   TYPE=FINAL
         END
```

```
M=KDATA

         PRINT NOGEN
MAND2    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
KDATA    DFHMDI   SIZE=(13,80),LINE=8,COLUMN=1
         DFHMDF   POS=(1,10),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='LOAN TYPE ?'
KLTYPE   DFHMDF   POS=(1,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,              *
               INITIAL='-'
         DFHMDF   POS=(1,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,43),ATTRB=(PROT,NORM),LENGTH=37,                 *
               INITIAL='(3 = EXTRA(MORTGAGE),4 = EXTRA(LAND))'
         DFHMDF   POS=(2,10),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='LAND AREA ?'
KAREA    DFHMDF   POS=(2,42),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,             *
               INITIAL='----'
         DFHMDF   POS=(2,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,10),ATTRB=(PROT,NORM),LENGTH=11,                 *
               INITIAL='EVAL/UNIT ?'
KUNIT    DFHMDF   POS=(3,40),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,             *
               INITIAL='------'
         DFHMDF   POS=(3,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,10),ATTRB=(PROT,NORM),LENGTH=15,                 *
               INITIAL='EVAL-BUILDING ?'
KBUILD   DFHMDF   POS=(4,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,             *
               INITIAL='-------'
         DFHMDF   POS=(4,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(5,10),ATTRB=(PROT,NORM),LENGTH=16,                 *
               INITIAL='FIRE INSURANCE ?'
KFIRE    DFHMDF   POS=(5,40),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,             *
               INITIAL='------'
         DFHMDF   POS=(5,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,10),ATTRB=(PROT,NORM),LENGTH=15,                 *
               INITIAL='EVALUATE-DATE ?'
KEDATE   DFHMDF   POS=(6,28),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,             *
               INITIAL='--'
         DFHMDF   POS=(6,31),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,33),ATTRB=(PROT,BRT),LENGTH=1,                   *
               INITIAL='/'
KEMON    DFHMDF   POS=(6,35),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,             *
               INITIAL='--'
         DFHMDF   POS=(6,38),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,40),ATTRB=(PROT,BRT),LENGTH=1,                   *
               INITIAL='/'
KEYEAR   DFHMDF   POS=(6,42),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,             *
               INITIAL='----'
         DFHMDF   POS=(6,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(7,10),ATTRB=(PROT,NORM),LENGTH=12,                 *
               INITIAL='EVAL-PRICE  '
KPRICE   DFHMDF   POS=(7,37),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,             *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(7,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,10),ATTRB=(PROT,NORM),LENGTH=27,                 *
               INITIAL='AMOUNT GUARANTEE SECURITIES'
GSECUR   DFHMDF   POS=(8,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,             *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(8,47),ATTRB=(ASKIP),LENGTH=1
LINE2    DFHMDF   POS=(12,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF   POS=(12,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
D=C.KDATA

         PRINT NOGEN
NLAND2   DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
KDATA    DFHMDI   SIZE=(13,80),LINE=8,COLUMN=1
         DFHMDF   POS=(1,10),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LOAN TYPE ?'
KLTYPE   DFHMDF   POS=(1,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,             *
               INITIAL='-'
         DFHMDF   POS=(1,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,43),ATTRB=(PROT,NORM),LENGTH=37,                *
               INITIAL='(3 = EXTRA(MORTGAGE),4 = EXTRA(LAND))'
         DFHMDF   POS=(2,10),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='LAND AREA ?'
KAREA    DFHMDF   POS=(2,42),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               INITIAL='----'
         DFHMDF   POS=(2,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,10),ATTRB=(PROT,NORM),LENGTH=11,                *
               INITIAL='EVAL/UNIT ?'
KUNIT    DFHMDF   POS=(3,40),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,            *
               INITIAL='------'
         DFHMDF   POS=(3,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(4,10),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='EVAL-BUILDING ?'
KBUILD   DFHMDF   POS=(4,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,            *
               INITIAL='-------'
         DFHMDF   POS=(4,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(5,10),ATTRB=(PROT,NORM),LENGTH=16,                *
               INITIAL='FIRE INSURANCE ?'
KFIRE    DFHMDF   POS=(5,40),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,            *
               INITIAL='------'
         DFHMDF   POS=(5,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,10),ATTRB=(PROT,NORM),LENGTH=15,                *
               INITIAL='EVALUATE-DATE ?'
KEDATE   DFHMDF   POS=(6,28),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,            *
               INITIAL='--'
         DFHMDF   POS=(6,31),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,33),ATTRB=(PROT,BRT),LENGTH=1,                  *
               INITIAL='/'
KEMON    DFHMDF   POS=(6,35),ATTRB=(UNPROT,BRT,NUM),LENGTH=2,            *
               INITIAL='--'
         DFHMDF   POS=(6,38),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(6,40),ATTRB=(PROT,BRT),LENGTH=1,                  *
               INITIAL='/'
KEYEAR   DFHMDF   POS=(6,42),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,            *
               INITIAL='----'
         DFHMDF   POS=(6,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(7,10),ATTRB=(PROT,NORM),LENGTH=12,                *
               INITIAL='EVAL-PRICE  '
KPRICE   DFHMDF   POS=(7,37),ATTRB=(UNPROT,BRT,NUM),LENGTH=9,            *
               PICOUT='Z,ZZZ,ZZ9'
         DFHMDF   POS=(7,47),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(8,10),ATTRB=(PROT,NORM),LENGTH=27,                *
               INITIAL='AMOUNT GUARANTEE SECURITIES'
GSECUR   DFHMDF   POS=(8,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=7,            *
               PICOUT='ZZZ,ZZ9'
         DFHMDF   POS=(8,47),ATTRB=(ASKIP),LENGTH=1
LINE2    DFHMDF   POS=(12,2),ATTRB=(UNPROT,BRT),LENGTH=76
         DFHMDF   POS=(12,79),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
%NEWEVA
         PRINT NOGEN
MLANDI   DFHMSD  TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,          *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
NEWEVA   DFHMDI  SIZE=(3,80),LINE=5,COLUMN=1
         DFHMDF  POS=(2,10),ATTRB=(PROT,NORM),LENGTH=19,                 *
               INITIAL='EVALUATE NEW LAND ?'
KEYA     DFHMDF  POS=(2,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,              *
               INITIAL='-'
         DFHMDF  POS=(2,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF  POS=(2,43),ATTRB=(PROT,NORM),LENGTH=16,                 *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD  TYPE=FINAL
                 END
```

```
D=C.NEWEVA
         PRINT NOGEN
*L AND1  DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
NEWEVA   DFHMDI   SIZE=(3,80),LINE=5,COLUMN=1
         DFHMDF   POS=(2,10),ATTRB=(PROT,NORM),LENGTH=19,            *
               INITIAL='EVALUATE NEW LAND ?'
KEYA     DFHMDF   POS=(2,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,         *
               INITIAL='-'
         DFHMDF   POS=(2,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,43),ATTRB=(PROT,NORM),LENGTH=16,            *
               INITIAL='(Y = YES,N = NO)'
         DFHMSD   TYPE=FINAL
                  END
```

```
¶=LGUA

         PRINT NOGEN
LGUA1    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LGUA     DFHMDI   SIZE=(3,90),LINE=21,COLUMN=1
         DFHMDF   POS=(1,6),ATTRB=(PROT,NORM),LENGTH=31,                *
               INITIAL='ACCEPT THIS LAND TO GUARANTEE ?'
ANSWER   DFHMDF   POS=(1,38),ATTRB=(UNPROT,BRT,IC),LENGTH=1,            *
               INITIAL='-'
         DFHMDF   POS=(1,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,43),ATTRB=(PROT,NORM),LENGTH=26,              *
               INITIAL='(Y = YES,N = NO)  APPLY# ?'
NOAPL    DFHMDF   POS=(1,72),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,          *
               INITIAL='----'
         DFHMDF   POS=(1,77),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,26),ATTRB=(PROT,NORM),LENGTH=11,              *
               INITIAL='ID-MEMBER ?'
IDMEM    DFHMDF   POS=(2,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(2,46),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,54),ATTRB=(PROT,NORM),LENGTH=15,              *
               INITIAL='AMOUNT TO USE ?'
AMTGS    DFHMDF   POS=(2,70),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(2,77),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,26),ATTRB=(PROT,NORM),LENGTH=11,              *
               INITIAL='USE SHARE ?'
LSHARE   DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(3,46),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,56),ATTRB=(PROT,NORM),LENGTH=13,              *
               INITIAL='USE DEPOSIT ?'
LDEPOS   DFHMDF   POS=(3,70),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(3,77),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
0=C.LGUA

         PRINT NOGEN
LGUA1    DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
LGUA     DFHMDI   SIZE=(3,80),LINE=21,COLUMN=1
         DFHMDF   POS=(1,6),ATTRB=(PROT,NORM),LENGTH=31,               *
               INITIAL='ACCEPT THIS LAND TO GUARANTEE ?'
ANSWER   DFHMDF   POS=(1,39),ATTRB=(UNPROT,BRT,IC),LENGTH=1,           *
               INITIAL='-'
         DFHMDF   POS=(1,41),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(1,43),ATTRB=(PROT,NORM),LENGTH=26,              *
               INITIAL='(Y = YES,N = NO)  APPLY# ?'
NOAPL    DFHMDF   POS=(1,72),ATTRB=(UNPROT,BRT,NUM),LENGTH=4,          *
               INITIAL='----'
         DFHMDF   POS=(1,77),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,26),ATTRB=(PROT,NORM),LENGTH=11,              *
               INITIAL='ID-MEMBER ?'
IDMEM    DFHMDF   POS=(2,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(2,46),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(2,54),ATTRB=(PROT,NORM),LENGTH=15,              *
               INITIAL='AMOUNT TO USE ?'
AMTGS    DFHMDF   POS=(2,70),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(2,77),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,26),ATTRB=(PROT,NORM),LENGTH=11,              *
               INITIAL='USE SHARE ?'
LSHARE   DFHMDF   POS=(3,39),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(3,46),ATTRB=(ASKIP),LENGTH=1
         DFHMDF   POS=(3,56),ATTRB=(PROT,NORM),LENGTH=13,              *
               INITIAL='USE DEPOSIT ?'
LDEPOS   DFHMDF   POS=(3,70),ATTRB=(UNPROT,BRT,NUM),LENGTH=6,          *
               INITIAL='------'
         DFHMDF   POS=(3,77),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
                  END
```

```
M=MAPERR1

         PRINT NOGEN
CMAP     DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,        *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1  DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
ERR      DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,              *
               JUSTIFY=(LEFT,BLANK)
         DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
         DFHMSD   TYPE=FINAL
         END
```

D=C.MAPERR1

```
          PRINT NOGEN
CMAP      DFHMSD   TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,TERM=3270,      *
               CTRL=(FREEKB,FRSET),DATA=FIELD,TIOAPFX=YES
MAPERR1   DFHMDI   SIZE=(1,80),LINE=24,COLUMN=1
LERR      DFHMDF   POS=(01,1),ATTRB=(UNPROT,BRT),LENGTH=69,          *
               JUSTIFY=(LEFT,BLANK)
          DFHMDF   POS=(01,71),ATTRB=(ASKIP),LENGTH=1
          DFHMSD   TYPE=FINAL
          END
```

ค.๒ โปรแกรมแสดงลักษณะงานที่ต้องการ (โปรแกรม P01)

```
ID DIVISION.
PROGRAM-ID. P01TITLE.
AUTHOR. KUNLAYANEE.
DATE-COMPILED.
REMARKS. DISPLAY INFORMATION OF ITEM.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
77  MESS-I                          PIC  S9(4)  COMP VALUE +0.
77  MESS-OUT                        PIC  X(60).
77  KEY9                            PIC  9(11).
77  KOPTION                         PIC  9(01)  VALUE 0.
01  NEW-DATE.
    02  N-YY                        PIC  9(04).
    02  N-MM                        PIC  9(02).
    02  N-DD                        PIC  9(02).
01  DATA-REC.
    02  ID-TRAN-KEY.
      03  YEAR-TR                   PIC  9(04).
      03  MONTH-TR                  PIC  9(02).
      03  APPLY-NO                  PIC  9(04).
      03  SEQUENCE-NO               PIC  9(01).
    02  IDTR                        PIC  9(06).
    02  L-TYPE-TR                   PIC  X(01).
    02  PRINCIPAL-TR                PIC  9(06).
    02  TOTAL-PERIOD-TR             PIC  9(03).
    02  ID-GUA-TR                   PIC  9(09).
    02  GUA-STAT-TR                 PIC  9(01).
    02  GUA-NUM-TR                  PIC  9(01).
    02  GUA-LIMIT-TR                PIC  9(06).
    02  GUA-SHARE-TR                PIC  9(06).
    02  GUA-DEPOSIT-TR              PIC  9(06).
    02  EVAL-DATE-TR                PIC  9(08).
    02  LAND-AREA-TR                PIC  9(04).
    02  EVAL-UNIT-TR                PIC  9(06).
    02  EVAL-BUILDING-TR            PIC  9(07).
    02  EVAL-INSURANCE-TR           PIC  9(06).
    02  UPDATE-DATE-TR              PIC  9(08).
    02  BLANK-TR                    PIC  X(33).
01  DFHAID      COPY DFHAID.
01  MTITLEI     COPY MTITLE.
PROCEDURE DIVISION.
P-START.
    EXEC CICS HANDLE CONDITION
              INVMPSZ (P-INVMPSZ)
              MAPFAIL (P-MAPFAIL)
              DSIDERR (P-DSIDERR)
              LENGERR (P-LENGERR)
              NOTFND (P-NOTFND)
              NOSTG   (P-NOSTG)
              PGMIDERR (P-PERR) END-EXEC.
    EXEC CICS HANDLE AID
              CLEAR (P-CLEAR)
              ANYKEY (P-ANYKEY)  END-EXEC.
    EXEC CICS SEND MAP ('MTITLE') MAPONLY FREEKB ERASE END-EXEC.
    EXEC CICS RECEIVE MAP ('MTITLE') INTO (MTITLEI) END-EXEC.
    MOVE KDATEI TO N-DD.
    MOVE KMONTHI TO N-MM.
    MOVE KYEARI TO N-YY.
    MOVE CHOSENI TO KOPTION.
P-READ.
    MOVE 90000000000 TO KEY9.
    EXEC CICS READ INTO (DATA-REC)
                   DATASET ('CIMAFL')
                   RIDFLD (KEY9)
                   UPDATE
                   KEYLENGTH (11) END-EXEC.
    MOVE NEW-DATE TO EVAL-DATE-TR.
    MOVE NEW-DATE TO UPDATE-DATE-TR.
    EXEC CICS REWRITE DATASET ('CIMAFL')
                      FROM (DATA-REC)
                      END-EXEC.
    IF KOPTION = 1
        EXEC CICS XCTL
             PROGRAM ('LOANDEST')
```

```
                END-EXEC.
    IF KOPTION = 2
       EXEC CICS XCTL
                PROGRAM ('PREPARE1')
                END-EXEC.
    IF KOPTION = 3
        EXEC CICS XCTL
              PROGRAM ('LOANGUAM')
                END-EXEC.
    IF KOPTION = 4
        EXEC CICS XCTL
                PROGRAM ('PLOANGTL')
                END-EXEC.
    MOVE 'OPTION CODE ERROR' TO LERRO.
    EXEC CICS SEND MAP ('MTITLE')
                FROM (MTITLEO)
                END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT')  END-EXEC.
P-CLEAR.
    EXEC CICS RETURN END-EXEC.
P-ANYKEY.
    MOVE 32 TO MESS-L.
    MOVE 'DO NOT PRESS ENTER KEY TRY AGAIN' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT') END-EXEC.
P-INVMPSZ.
    MOVE 16 TO MESS-L.
    MOVE 'INVALID MAP SIZE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT')  END-EXEC.
P-MAPFAIL.
    MOVE 8 TO MESS-L.
    MOVE 'MAP FAIL' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT')  END-EXEC.
P-DSIDERR.
    MOVE 17 TO MESS-L.
    MOVE 'DATA SET ID ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT')  END-EXEC.
P-LENGERR.
    MOVE 18 TO MESS-L.
    MOVE 'WRONG LENGTH ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT')  END-EXEC.
P-NOTFND.
    MOVE 16 TO MESS-L.
    MOVE 'RECORD NOT FOUND' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT') END-EXEC.
P-NOSTG.
    MOVE 10 TO MESS-L.
    MOVE 'NO STORAGE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PTIT') END-EXEC.
P-PERR.
    MOVE 13 TO MESS-L.
    MOVE 'PROGRAM ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
    STOP RUN.
```

ค.๓ โปรแกรมแสดงรายละเอียดการกู้ของสมาชิก (โปรแกรม P02)

```
ID DIVISION.
PROGRAM-ID. LOANDEST.
AUTHOR. SAVING.
DATE-COMPILED.
REMARKS. DISPLAY OLD LOAN AND CHECK GUARANTOR.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
77  MESS-L                          PIC  S9(4)  COMP VALUE +0.
77  MESS-OUT                        PIC  X(75).
77  MARRY-ID                        PIC  9(06).
77  C-REMAIN                        PIC  S9(06) VALUE +0.
77  CGUA1                           PIC  9(09)  VALUE 0.
77  CGUA2                           PIC  9(09)  VALUE 0.
77  CGUA3                           PIC  9(09)  VALUE 0.
77  CGUA4                           PIC  9(09)  VALUE 0.
77  CGUA5                           PIC  9(09)  VALUE 0.
77  GUAL                            PIC  9(08)  VALUE 0.
77  GUAL1                           PIC  9(07)  VALUE 0.
77  GUAL2                           PIC  9(07)  VALUE 0.
77  GUAL3                           PIC  9(07)  VALUE 0.
77  GUAL4                           PIC  9(07)  VALUE 0.
77  GUAL5                           PIC  9(07)  VALUE 0.
77  EMP                             PIC  9(04)  VALUE 2500.
77  EXP                             PIC  9(06)  VALUE 300000.
77  P-AVERAGE                       PIC  9(06)  VALUE 0.
77  KLTYPE                          PIC  X(01)  VALUE SPACE.
77  REDEBT1                         PIC  9(04)  VALUE 0.
77  CHOSE1                          PIC  X(01)  VALUE SPACE.
77  SHADEP                          PIC  9(07)  VALUE 0.
77  KPRINC                          PIC  9(06)  VALUE 0.
77  KPERID                          PIC  9(03)  VALUE 0.
77  KAMT                            PIC  9(06)  VALUE 0.
77  KEEPG                           PIC  9(07)  VALUE 0.
77  WQUEUE                          PIC  9(02)  VALUE 0.
77  REDEBT                          PIC  9(06)  VALUE 0.
77  TOT-GUA                         PIC  9(06)  VALUE 0.
77  DATAKEY                         PIC  9(11)  VALUE 0.
01  DEMAID       COPY DEMAID.
01  MEMB-REC.
    02  ID-KEY-M.
        03  ID-MEMBER-M             PIC  9(06).
        03  A-I-M                   PIC  X(01).
    02  CHECK-M.
        03  P-T-M                   PIC  X(01).
        03  REC-STATUS-M            PIC  X(01).
        03  F-L-M                   PIC  X(01).
    02  ID-MEMBER-OLD               PIC  9(04).
    02  NAME-THAI                   PIC  X(60).
    02  NAME-ENG                    PIC  X(40).
    02  SEX                         PIC  X(01).
    02  PAY-PLACE                   PIC  9(03).
    02  ID-PAY                      PIC  X(07).
    02  ID-OFFICE                   PIC  X(07).
    02  RE-MEMBER                   PIC  X(01).
    02  NAME-COND                   PIC  X(11).
    02  UPDATE-DATE-M               PIC  X(08).
    02  BLANK-M                     PIC  X(18).
01  STAT-REC.
    02  ID-KEY-S.
        03  ID-MEMBER-S             PIC  9(06).
        03  A-I-S                   PIC  X(01).
    02  CHECK-S.
        03  P-T-S                   PIC  X(01).
        03  REC-STATUS-S            PIC  X(01).
        03  F-L-S                   PIC  X(01).
    02  MEMBER-DATE                 PIC  9(08).
    02  MEMBER-AGE                  PIC  9(03).
    02  MARITAL                     PIC  X(01).
    02  ID-SPOUSE                   PIC  9(06).
    02  DEPT-CODE                   PIC  X(03).
    02  OFFICE-CODE                 PIC  X(03).
    02  MEMBER-TYPE                 PIC  X(01).
```

```
    02  L-QUEUE                    PIC  9(02).
    02  AMT-LOAN                   PIC  9(05).
    02  L-STATUS.
        03  EMERGENCY              PIC  X(01).
        03  ORDINARY               PIC  X(01).
        03  EXTRA                  PIC  X(01).
    02  AMT-GUA                    PIC  9(05).
    02  GUA-LIMIT-USED             PIC  9(05).
    02  GUA-MEN-STATUS             PIC  9(01).
    02  GUA-MEN-RECORD             PIC  9(01).
    02  SALARY                     PIC  9(05).
    02  O-INCOME                   PIC  9(04).
    02  XF-PRES-S                  PIC  9(04).
    02  O-PAYMENT                  PIC  9(05)V99.
    02  M-ADJUST                   PIC  9(03)V99.
    02  OUT-COND                   PIC  X(01).
    02  OUT-DATE                   PIC  X(08).
    02  UPDATE-DATE-S              PIC  X(08).
    02  BLANK-S                    PIC  X(29).
01  LOAN-REC.
    02  ID-KEY-L.
        03  ID-MEMBER-L            PIC  9(06).
        03  L-TYPE-L               PIC  X(01).
        03  A-I-L1                 PIC  X(01).
    02  CHECK-L.
        03  P-T-L1                 PIC  X(01).
        03  REC-STATUS-L           PIC  X(01).
        03  F-L-L                  PIC  X(01).
    02  L-AGREE-NO                 PIC  9(07).
    02  L-DATE                     PIC  9(08).
    02  PRINCIPAL-L                PIC  9(06).
    02  L-FLAG                     PIC  X(01).
    02  PRING-F-L-PAID             PIC  9(05).
    02  PRINC-MONTHLY-PAID         PIC  9(05).
    02  DEBT-REMAIN                PIC  9(06).
    02  TOTAL-PERIOD               PIC  9(03).
    02  PERIOD                     PIC  9(03).
    02  INT-F-L-PAID               PIC  9(04)V99.
    02  INT-PRES-PAID              PIC  9(04)V99.
    02  INT-DECRE                  PIC  9(03)V99.
    02  INT-RATE                   PIC  9(02)V9.
    02  INTX                       PIC  9(05)V99.
    02  GUA-MEN-KEY-L.
        03  ID-GUA-MEN-L           PIC  9(06).
        03  SEQUENCE-L2            PIC  9(01).
    02  GUA-LAND-KEY-L.
        03  ID-GUA-LAND-L          PIC  9(09).
        03  SEQUENCE-L3            PIC  9(01).
    02  GUA-LAND-NUMB              PIC  9(01).
    02  GUA-SHARE                  PIC  9(06).
    02  GUA-DEPOSIT                PIC  9(06).
    02  UPDATE-DATE-L              PIC  X(08).
    02  BLANK-L                    PIC  X(08).
01  PERS-GUA-REC.
    02  GUA-MEN-KEY-P.
        03  ID-GUA-MEN-P           PIC  9(06).
        03  SEQUENCE-P1            PIC  9(01).
        03  A-I-P1                 PIC  X(01).
    02  CHECK-P.
        03  P-T-P                  PIC  X(01).
        03  REC-STATUS-P           PIC  X(01).
        03  F-L-P                  PIC  X(01).
    02  ID-GUAED                   PIC  9(06).
    02  GUA-LIMIT                  PIC  9(05).
    02  ID-CO-GUA-KEY.
        03  ID-CO-GUA              PIC  9(06).
        03  SEQUENCE-P2            PIC  9(01).
    02  UPDATE-DATE-P              PIC  X(08).
    02  BLANK-P                    PIC  X(27).
01  ASSET-GUA-REC.
    02  GUA-LAND-KEY-A.
        03  ID-GUA-LAND-A          PIC  9(09).
        03  SEQUENCE-A1            PIC  9(01).
        03  A-I-A1                 PIC  X(01).
    02  CHECK-A.
        03  P-T-A                  PIC  X(01).
        03  REC-STATUS-A           PIC  X(01).
        03  F-L-A                  PIC  X(01).
```

```
    02  L-TYPE-A                    PIC  X(01).
    02  ID-MEMBER-A                 PIC  9(06).
    02  PRINCIPAL-A                 PIC  9(06).
    02  GUA-LIMIT-A                 PIC  9(06).
    02  GUA-LAND-CO-KEY.
        03  ID-GUA-LAND-CO          PIC  9(09).
        03  SEQUENCE-A2             PIC  9(01).
    02  UPDATE-DATE-A               PIC  X(08).
    02  BLANK-A                     PIC  X(13).
01  LAND-STAT-REC.
    02  GUA-LAND-KEY-D.
        03  ID-GUA-LAND-D           PIC  9(09).
        03  A-I-D                   PIC X(01).
    02  CHECK-D.
        03  P-T-D                   PIC  X(01).
        03  REC-STATUS-D            PIC  X(01).
        03  F-L-D                   PIC  X(01).
    02  L-TYPE-D                    PIC  X(01).
    02  GUA-LAND-RECORD             PIC  9(01).
    02  MEN-PER-LAND                PIC  9(01).
    02  EVAL-DATE                   PIC  9(08).
    02  LAND-AREA                   PIC  9(04).
    02  EVAL-UNIT                   PIC  9(06).
    02  EVAL-PRICE                  PIC  9(07).
    02  EVAL-BUILDING               PIC  9(07).
    02  FIRE-INSURANCE              PIC  9(06).
    02  AMT-GUA-SECUR               PIC  9(06).
    02  REMAIN-SECUR                PIC  9(06).
    02  UPDATE-DATE-D               PIC  X(08).
    02  BLANK-D                     PIC  X(54).
01  DATA-REC.
    02  ID-TRAN-KEY.
      03  YEAR-TR                   PIC  9(04).
      03  MONTH-TR                  PIC  9(02).
      03  APPLY-NO                  PIC  9(04).
      03  SEQUENCE-NO               PIC  9(01).
    02  ID-TR                       PIC  9(06).
    02  L-TYPE-TR                   PIC  X(01).
    02  PRINCIPAL-TR                PIC  9(06).
    02  TOTAL-PERIOD-TR             PIC  9(03).
    02  ID-GUA-TR                   PIC  9(09).
    02  GUA-STAT-TR                 PIC  9(01).
    02  GUA-NUM-TR                  PIC  9(01).
    02  GUA-LIMIT-TR                PIC  9(06).
    02  GUA-SHARE-TR                PIC  9(06).
    02  GUA-DEPOSIT-TR              PIC  9(06).
    02  EVAL-DATE-TR                PIC  9(08).
    02  LAND-AREA-TR                PIC  9(04).
    02  EVAL-UNIT-TR                PIC  9(06).
    02  EVAL-BUILDING-TR            PIC  9(07).
    02  EVAL-INSURANCE-TR           PIC  9(06).
    02  UPDATE-DATE-TR              PIC  9(08).
    02  BLANK-TR                    PIC  X(33).
01  STAT-KEY.
    02  ID-S                        PIC  9(06).
    02  ACTIVE                      PIC  X(01).
01  P-DATE.
    02  DD1                         PIC  9(02).
    02  FILLER                      PIC  X(01)  VALUE '/'.
    02  MM1                         PIC  9(02).
    02  FILLER                      PIC  X(01)  VALUE '/'.
    02  YY1                         PIC  9(04).
01  LOAN-STAT.
    02  L-EMERGENCY                 PIC  X(01).
    02  L-ORDINARY                  PIC  X(01).
    02  L-EXTRA                     PIC  X(01).
01  LOAN-KEY.
    02  ID-L                        PIC  9(06).
    02  TYPE-L                      PIC  X(01).
    02  ACTIVE-L                    PIC  X(01).
01  LAND-KEY.
    02  LAND-ID                     PIC  9(09).
    02  LAND-ACT                    PIC  X(01).
01  ASSET-KEY.
    02  ASSET-ID.
        03  A-ID                    PIC  9(09).
        03  A-SEQ                   PIC  9(01).
```

```
           02  ASSET-ACT                  PIC  X(01).
       01  PERS-KEY.
           02  PERS-ID.
               03  P-ID                   PIC  9(06).
               03  P-SEQ                  PIC  9(01).
           02  PERS-ACT                   PIC X(01).
       01  KGUA.
           02  PRE-G                      PIC  9(03).
           02  LAST-G                     PIC  9(06).
       01  DATE-1.
           02  YY-1                       PIC  9(04).
           02  MM-1                       PIC  9(02).
           02  DD-1                       PIC  9(02).
       01  LMAPPI      COPY LMAPP.
       01  LDESCTI     COPY LDESCT.
       01  INQYLI      COPY INQYL.
       01  MAPERR1I    COPY MAPERR1.
       PROCEDURE DIVISION.
       P-START.
           EXEC CICS HANDLE CONDITION
                     INVMPSZ (P-INVMPSZ)
                     MAPFAIL (P-MAPFAIL)
                     DSIDERR (P-DSIDERR)
                     LENGERR (P-LENGERR)
                     NOTFND  (P-NOTFND)
                     NOSTG   (P-NOSTG)  END-EXEC.
           EXEC CICS HANDLE AID
                     CLEAR (P-CLEAR)
                     ANYKEY (P-ANYKEY)  END-EXEC.
           MOVE '0000000000' TO DATAKEY.
           EXEC CICS READ INTO (DATA-REC)
                          DATASET ('CIMAFL')
                          RIDFLD (DATAKEY)
                          KEYLENGTH (11) END-EXEC.
           MOVE EVAL-DATE-TR TO DATE-1.
           MOVE DD-1 TO DD1.
           MOVE MM-1 TO MM1.
           MOVE YY-1 TO YY1.
           MOVE P-DATE TO KDATEO LDATEO.
       P-INPUT.
           MOVE LOW-VALUE TO LDESCTO.
           MOVE LOW-VALUE TO MAPERR1O.
           EXEC CICS SEND MAP ('LMAPP') MAPONLY FREEKB ERASE END-EXEC.
           EXEC CICS RECEIVE MAP ('LMAPP') INTO (LMAPPI) END-EXEC.
           MOVE KEYLNI TO ID-S KIDMEMO.
           MOVE '0' TO ACTIVE.
           EXEC CICS READ INTO (MEMB-REC)
                     DATASET ('MEMBFL')
                     RIDFLD (STAT-KEY)
                     KEYLENGTH (7)  END-EXEC.
           MOVE NAME-ENG TO KNAMEO  NAMEDO.
           EXEC CICS SEND MAP ('LMAPP') FROM (LMAPPO) END-EXEC.
      *****READ STAT FILE****
       P-READF2.
           EXEC CICS READ INTO (STAT-REC)
                     DATASET ('STATFL')
                     RIDFLD (STAT-KEY)
                     KEYLENGTH (7)  END-EXEC.
       P-READF2-EXIT. EXIT.
       CHECK-QUEUE.
           IF CHECK-S = '000' OR '110' GO TO KEEP-LSTATUS.
           GO TO ERR-REC2.
       P-READF2I.
           IF CHECK-S = '110' NEXT SENTENCE
           ELSE GO TO ERR-REC2.
           IF CHOSE1 = 'Y' MOVE '1' TO ACTIVE
                           PERFORM P-READF2 THRU P-READF2-EXIT
                           IF CHECK-S = '100' GO TO KEEP-LSTATUS
                           ELSE GO TO  ERR-REC2I.
       KEEP-LSTATUS.
           MOVE ID-SPOUSE TO IDMARYO MARRY-ID.
           MOVE L-STATUS TO LOAN-STAT.
           MOVE L-QUEUE TO WQUEUE.
           IF L-EMERGENCY = '0'
               MOVE SPACE TO TORP1O
               MOVE ZERO TO OTYPE1O OPRINC1O  OPERIOD1O OBERT1O.
```

```
     IF L-EMERGENCY = '1' PERFORM OLD-LL THRU P-MOVEF4I-EXIT.
     IF L-ORDINARY = '0' AND L-EXTRA = '0'
         MOVE '2' TO OTYPEO
         MOVE SPACE TO TORP2O
         MOVE ZERO TO SDEPO
         MOVE ZERO TO OPRINCO OPERIDO OGUA1O OAMTG1O OREM1O ODEBTO
                                        OGUA2O OAMTG2O OREM2O
                                        OGUA3O OAMTG3O OREM3O
                                        OGUA4O OAMTG4O OREM4O
                                        OGUA5O OAMTG5O OREM5O
         GO TO P-NEWLOAN.
     IF L-ORDINARY = '2' GO TO OLD-L2.
*****************************************************************
*****OLD EXTRA LOAN****
*****************************************************************

     MOVE ID-S TO ID-L.
     MOVE L-EXTRA TO TYPE-L.
     MOVE '0' TO ACTIVE-L.
 P-READF4.
     EXEC CICS READ INTO (LOAN-REC)
               DATASET ('LOANFL')
               RIDFLD (LOAN-KEY)
               KEYLENGTH (8)  END-EXEC.
 P-READF4-EXIT. EXIT.
*****************************************************************
****CHECK LOAN FILE STATUS****
*****************************************************************

 P-CHECKL.
     IF CHOSE1 = 'Y' AND CHECK-L = '001' MOVE 'PREV' TO TORP2O
                                         GO TO P-MOVEF4.
     IF CHOSE1 = 'Y' AND CHECK-L = '110'
         MOVE '1' TO ACTIVE-L
         PERFORM P-READF4 THRU P-READF4-EXIT
         IF CHECK-L = '100' MOVE  'PREV' TO TORP2O
                            GO TO P-MOVEF4
         ELSE GO TO ERR-REC4I.
     IF CHOSE1 NOT = 'Y' AND CHECK-L = '000' MOVE 'PERM' TO TORP2O
                                             GO TO P-MOVEF4.
     IF CHOSE1 NOT = 'Y' AND CHECK-L = '110'
         MOVE 'TEMP' TO TORP2O.
 P-MOVEF4.
     MOVE L-TYPE-L TO OTYPEO.
     MOVE PRINCIPAL-L TO OPRINCO.
     MOVE PERIOD TO OPERIDO.
     MOVE DEBT-REMAIN TO REDEBT ODEBTO.
 P-MOVEF4-EXIT. EXIT.
 P-EXTRA.
     MOVE GUA-SHARE TO SDEPIO.
     MOVE GUA-DEPOSIT TO SDEPO.
     MOVE GUA-LAND-KEY-L TO ASSET-IO.
     MOVE IO-GUA-LAND-L TO OGUA1O LAND-IO GGUA1.
     MOVE '0' TO LAND-ACT.
 P-READF7.
     EXEC CICS READ INTO (LAND-STAT-REC)
               DATASET ('LANDFL')
               RIDFLD (LAND-KEY)
               KEYLENGTH (10)  END-EXEC.
 P-READF7-EXIT. EXIT.
 CHECK-APDF71.
     IF CHECK-D = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC7.
 P-MOVEGUAL.
     MOVE REMAIN-SECUR TO OREM1O.
 P-READF6.
     MOVE '0' TO ASSET-ACT.
     EXEC CICS READ INTO (ASSET-GUA-REC)
               DATASET ('ASSTFL')
               RIDFLD (ASSET-KEY)
               KEYLENGTH (11) END-EXEC.
     IF CHECK-A = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC6.
     IF ID-MEMBER-A NOT = ID-L GO TO ERR-GUAL.
     MOVE GUA-LAND-CO-KEY TO ASSET-ID.
 P-EXIT1.
     EXIT.
```

```
P-CKD.
     MOVE GUA-LIMIT-A TO OAMTG1D GUAL1.
     IF A-ID = 0 GO TO P-ZERO4.
     MOVE A-ID TO LAND-ID.
     MOVE '0' TO LAND-ACT.
     PERFORM P-READF7 THRU P-READF7-EXIT.
*****CHECK GUA2****
     IF CHECK-D = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC7.
     MOVE ID-GUA-LAND-D TO OGUA2D CGUA2.
     MOVE REMAIN-SECUR TO OREM2D.
     PERFORM P-READF6 THRU P-EXIT1.
     MOVE GUA-LIMIT-A TO OAMTG2D GUAL2.
     IF A-ID = 0 GO TO P-ZERO3.
*****CHECK GUA3****
     MOVE A-ID TO LAND-ID.
     MOVE '0' TO LAND-ACT.
     PERFORM P-READF7 THRU P-READF7-EXIT.
     IF CHECK-D = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC7.
     MOVE ID-GUA-LAND-D TO OGUA3D CGUA3.
     MOVE REMAIN-SECUR TO OREM3D.
     PERFORM P-READF6 THRU P-EXIT1.
     MOVE GUA-LIMIT-A TO OAMTG3D GUAL3.
     IF A-ID = 0 GO TO P-ZERO2.
*****CHECK GUA4****
     MOVE A-ID TO LAND-ID.
     MOVE '0' TO LAND-ACT.
     PERFORM P-READF7 THRU P-READF7-EXIT.
     IF CHECK-D = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC7.
     MOVE ID-GUA-LAND-D TO OGUA4D CGUA4.
     MOVE REMAIN-SECUR TO OREM4D.
     PERFORM P-READF6 THRU P-EXIT1.
     MOVE GUA-LIMIT-A TO OAMTG4D GUAL4.
     IF A-ID = 0 GO TO P-ZERO1.
*****CHECK GUA5****
     MOVE A-ID TO LAND-ID.
     MOVE '0' TO LAND-ACT.
     PERFORM P-READF7 THRU P-READF7-EXIT.
     IF CHECK-D = '000' OR '110' NEXT SENTENCE
     ELSE GO TO ERR-REC7.
     MOVE ID-GUA-LAND-D TO OGUA5D CGUA5.
     PERFORM P-READF6 THRU P-EXIT1.
     MOVE GUA-LIMIT-A TO OAMTG5D GUAL5.
     MOVE REMAIN-SECUR TO OREM5D.
     GO TO P-NEWLOAN.
P-ZERO4.
     MOVE 0 TO OGUA2D OAMTG2D OREM2D
               OGUA3D OAMTG3D OREM3D
               OGUA4D OAMTG4D OREM4D
               OGUA5D OAMTG5D OREM5D.
     GO TO P-NEWLOAN.
P-ZERO3.
     MOVE 0 TO OGUA3D OAMTG3D OREM3D
               OGUA4D OAMTG4D OREM4D
               OGUA5D OAMTG5D OREM5D.
     GO TO P-NEWLOAN.
P-ZERO2.
     MOVE 0 TO OGUA4D OAMTG4D OREM4D
               OGUA5D OAMTG5D OREM5D.
     GO TO P-NEWLOAN.
P-ZERO1.
     MOVE 0 TO OGUA5D OAMTG5D OREM5D.
     GO TO P-NEWLOAN.
**********************************************************************
****OLD EMERGENCY LOAN****
**********************************************************************

OLD-L1.
     MOVE ID-S TO ID-L.
     MOVE '1' TO TYPE-L.
     MOVE '0' TO ACTIVE-L.
     PERFORM P-READF4 THRU P-READF4-EXIT.
     IF CHOSE1 = 'Y' AND CHECK-L = '000' MOVE 'PREV' TO TORP1D
                                         GO TO P-MOVEF41.
```

```
      IF CHOSE1 = 'Y' AND CHECK-L = '110'
          MOVE '1' TO ACTIVE-L
          PERFORM P-READF4 THRU P-READF4-EXIT
          IF CHECK-L = '100' MOVE 'PREV' TO TOPP1O
                             GO TO P-MOVEF41
          ELSE GO TO ERR-REC41.
      IF CHOSE1 NOT = 'Y' AND CHECK-L = '110' MOVE 'PERM' TO TOPP1O
                                              GO TO P-MOVEF41.
      IF CHOSE1 NOT = 'Y' AND CHECK-L = '110'
          MOVE 'TEMP' TO TOPP1O.
  P-MOVEF41.
      MOVE L-TYPE-L TO OTYPE1O.
      MOVE PRINCIPAL-L TO OPRINC1O.
      MOVE PERIOD TO OPERID1O.
      MOVE DEBT-REMAIN TO ODEBT1O REDEBT1.
  P-MOVEF41-EXIT. EXIT.
*****************************************************************
*****OLD ORDINARY LOAN****
*****************************************************************
  OLO-L2.
      MOVE ID-S TO ID-L.
      MOVE 1-ORDINARY TO TYPE-L.
      MOVE '0' TO ACTIVE-L.
      PERFORM P-READF4 THRU P-MOVEF4-EXIT.
      MOVE GUA-SHARE TO SOEP1O.
      MOVE GUA-DEPOSIT TO SUEPO.
      MOVE GUA-MEM-KEY-L TO PERS-ID  KEEPG.
      IF KEEPG = 0 GO TO P-NOGUA.
      MOVE P-ID TO OGUA1O CGUA1.
  P-READF5.
      MOVE KEEPG TO PERS-ID.
      MOVE '0' TO PERS-ACT.
      EXEC CICS READ INTO (PERS-GUA-REC)
                DATASET ('PERSFL')
                RIDFID (PERS-KEY)
                KEYLENGTH (8)  END-EXEC.
      IF CHECK-P = '000' OR '110' NEXT SENTENCE
      ELSE GO TO ERR-REC5.
      IF ID-GUARD NOT = ID-L GO TO ERR-GUAM.
      MOVE ID-CO-GUA-KEY TO KEEPG.
  P-READF5-EXIT. EXIT.
  P-AMT1.
      MOVE GUA-LIMIT TO OAMTG1O GUAL1.
  P-READF2GUA.
      IF P-ID > 900000 MOVE ZERO TO C-REMAIN
                       GO TO P-READF2GUA-EXIT.
      MOVE P-ID TO ID-S
      MOVE '0' TO ACTIVE.
      PERFORM P-READF2 THRU P-READF2-EXIT.
*****SHOW GUA1****
      IF CHECK-S = '000' OR '110' NEXT SENTENCE
      ELSE GO TO ERR-REC2.
      COMPUTE C-REMAIN = (AMT-GUA - GUA-LIMIT-USED).
  P-READF2GUA-EXIT. EXIT.
  P-CONTD.
      MOVE C-REMAIN TO OREM1O.
      IF KEEPG = 0 GO TO P-ZERO4.
      PERFORM P-READF5 THRU P-READF5-EXIT.
*****SHOW GUA2****
      MOVE ID-GUA-MEM-P TO OGUA2O CGUA2.
      MOVE GUA-LIMIT TO OAMTG2O GUAL2.
      PERFORM P-READF2GUA THRU P-READF2GUA-EXIT.
      MOVE C-REMAIN TO OREM2O.
      IF KEEPG = 0 GO TO P-ZERO3.
      PERFORM P-READF5 THRU P-READF5-EXIT.
*****SHOW GUA3****
      MOVE ID-GUA-MEM-P TO OGUA3O CGUA3.
      MOVE GUA-LIMIT TO OAMTG3O GUAL3.
      PERFORM P-READF2GUA THRU P-READF2GUA-EXIT.
      MOVE C-REMAIN TO OREM3O
      IF KEEPG = 0 GO TO P-ZERO2.
      PERFORM P-READF5 THRU P-READF5-EXIT.
```

```
*****SHOW GUA4****
     MOVE ID-GUA-MEN-P TO CGUA40 CGUA4.
     MOVE GUA-LIMIT TO DAMTG40 GUAL4.
     PERFORM P-READF2GUA THRU P-READF2GUA-EXIT.
     MOVE C-REMAIN TO OREM40.
     IF KEEPG = 0 GO TO P-ZERO1.
     PERFORM P-READF5 THRU P-READF5-EXIT.
*****SHOW GUA5****
     MOVE ID-GUA-MEN-P TO CGUA50 CGUA5.
     MOVE GUA-LIMIT TO DAMTG50 GUAL5.
     PERFORM P-READF2GUA THRU P-READF2GUA-EXIT.
     MOVE C-REMAIN TO OREM50.
**********************************************************************
*****SEND MAP AND KEY NEW LOAN****
**********************************************************************
 P-NEWLOAN.
     MOVE LOW-VALUE TO INQYLO.
     EXEC CICS SEND MAP ('LDESCT')
                    FROM (LDESCTO)
                    FREEKB
                    END-EXEC.
 P-RECEIVE.
     EXEC CICS RECEIVE MAP ('LDESCT')
                       INTO (LDESCTI)
                       END-EXEC.
     IF PREDI = 'Y' MOVE ID-L TO ID-S
                    MOVE PREDI TO CHOSE1
                     MOVE '0' TO ACTIVE
                     PERFORM P-READF2 THRU P-READF2-EXIT
                     IF CHECK-S = '000' GO TO P-WANTLOAN
                     ELSE GO TO P-READF2I.
     IF PREDI = 'N' AND CHOSE1 = 'Y'
         MOVE SPACE TO CHOSE1
         MOVE ID-L TO ID-S
         MOVE '2' TO ACTIVE
         GO TO P-READF2.
     MOVE SPACE TO LERRO.
     PERFORM P-MERR THRU P-MERR-EXIT.
 P-WANTLOAN.
     MOVE ALL '=' TO LINE1O.
     EXEC CICS SEND MAP ('INQYL') FROM (INQYLO) FREEKB END-EXEC.
 P-REC1.
     EXEC CICS RECEIVE MAP ('INQYL') INTO (INQYLI) END-EXEC.
     MOVE NTYPE1 TO KLTYPE.
     IF KLTYPE = '9' GO TO P-STOP1.
     MOVE NPRINC1 TO KPRINC.
     MOVE NPERID1 TO KPERID.
     MOVE NGUA1 TO KGUA.
     MOVE NAMTG1 TO KAMT.
     IF WQUEUE NOT = 0 GO TO P-NOTLOAN.
     IF KPRINC < REDEBT GO TO P-NOLOAN.
     IF KLTYPE = '1'
         IF KPRINC > EMP GO TO P-APM
         ELSE  COMPUTE P-AVERAGE = KPRINC / 2
           IF P-AVERAGE > SALARY GO TO P-AVM
           ELSE MOVE 'ALLOW TO LOAN' TO LERRO
           PERFORM P-MERR THRU P-MERR-EXIT
           GO TO P-WANTLOAN.
     IF KLTYPE NOT = '2' GO TO P-NEXL.
     IF KPRINC > AMT-LOAN GO TO P-APM.
     IF KPERID > 48 MOVE 'TOTAL PERIOD IS OVER THAN 48 PERIODS'
     TO LERRO
         PERFORM P-MERR THRU P-MERR-EXIT
         GO TO P-WANTLOAN.
     COMPUTE P-AVERAGE = KPRINC / KPERID.
     IF P-AVERAGE > SALARY GO TO P-AVM.
     MOVE LAST-G TO ID-S.
     MOVE '0' TO ACTIVE.
     PERFORM P-READF2 THRU P-READF2-EXIT.
     IF CHECK-S = '000' OR '110' GO TO P-REMAIN.
     GO TO ERR-REC2.
**********************************************************************
*****COMPUTE AMT-REMAIN****
**********************************************************************
 P-REMAIN.
     COMPUTE C-REMAIN = (AMT-GUA - GUA-LIMIT-USED).
*****CHECK OLD GUARANTOR****
 P-12345.
     IF KGUA = CGUA1 OR CGUA2 OR CGUA3 OR CGUA4 OR CGUA5
         GO TO P-CHECK.
     COMPUTE C-REMAIN = (C-REMAIN - KAMT).
```

```
           MOVE C-REMAIN TO NREMO.
           MOVE SPACE TO LERRO.
           IF C-REMAIN < ZERO GO TO P-NO.
       P-YES.
           PERFORM P-ALLOW THRU P-ALLOW-EXIT.
           GO TO P-WANTLOAN.
       P-NO.
                PERFORM P-NOALLOW THRU P-NOALLOW-EXIT
                GO TO P-WANTLOAN.
      *****USE OLD GUA****
       P-CHECK.
           MOVE 0 TO GUAL.
           IF KGUA = CGUA1
               COMPUTE GUAL = GUAL1 + C-REMAIN
               GO TO COMPARE-GUAL.
           IF KGUA = CGUA2
               COMPUTE GUAL = GUAL2 + C-REMAIN
               GO TO COMPARE-GUAL.
           IF KGUA = CGUA3
               COMPUTE GUAL = GUAL3 + C-REMAIN
               GO TO COMPARE-GUAL.
           IF KGUA = CGUA4
               COMPUTE GUAL = GUAL4 + C-REMAIN
               GO TO COMPARE-GUAL.
           COMPUTE GUAL = GUAL5 + C-REMAIN.
       COMPARE-GUAL.
           COMPUTE C-REMAIN = (GUAL - KAMT).
           MOVE C-REMAIN TO NREMO.
           IF GUAL < KAMT GO TO P-NO
           ELSE GO TO P-YES.
      ************************************************************
      *****NEW EXTRA LOAN****
      ************************************************************
       P-NEXL.
           IF KPRINC > EXP GO TO P-APM.
           IF KPERID > 120
               MOVE 'TOTAL PERIOD IS OVER THAN 120 PERIODS' TO LERRO
               PERFORM P-MERR THRU P-MERR-EXIT
               GO TO P-WANTLOAN.
           COMPUTE P-AVERAGE = KPRINC / KPERID.
           MOVE NGUAI TO LAND-ID.
           MOVE '0' TO LAND-ACT.
           PERFORM P-READF7 THRU P-READF7-EXIT.
           IF CHECK-D = '000' OR '110' NEXT SENTENCE
           ELSE GO TO ERR-REC7.
           IF KLTYPE = 4 AND L-TYPE-D = 3
               PERFORM P-NOALLOW THRU P-NOALLOW-EXIT
           GO TO P-WANTLOAN.
           IF KLTYPE = 3 OR 4
               MOVE REMAIN-SECUR TO C-REMAIN
               GO TO P-12345.
           PERFORM P-NOALLOW THRU P-NOALLOW-EXIT.
           GO TO P-WANTLOAN.
       P-APM.
           MOVE 'NOT ALLOW TO LOAN BECAUSE LOAN APPLY IS MORE THAN LOAN
      -         'RIGHT' TO LERRO.
           PERFORM P-MERR THRU P-MERR-EXIT.
           GO TO P-WANTLOAN.
       P-AVM.
           MOVE 'NOT ALLOW TO LOAN BECAUSE MONTHLY-PAID IS MORECTHAN SAL
      -         'ARY' TO LERRO.
           PERFORM P-MERR THRU P-MERR-EXIT.
           GO TO P-WANTLOAN.
       P-MERR.
           EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
       P-MERR-EXIT. EXIT.
       P-CLEAR.
           EXEC CICS RETURN  END-EXEC.
       P-INVMPSZ.
           MOVE 16 TO MESS-L.
           MOVE 'INVALID MAP SIZE' TO MESS-OUT.
           EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                               ERASE WAIT END-EXEC.
           GO TO P-STOP.
```

```
P-MAPFAIL.
    MOVE 8 TO MESS-L.
    MOVE 'MAP FAIL' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    GO TO P-STOP.
P-DSIDERR.
    MOVE 17 TO MESS-L.
    MOVE 'DATA SET ID ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    GO TO P-STOP.
P-LENGERR.
    MOVE 18 TO MESS-L.
    MOVE 'WRONG LENGTH ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    GO TO P-STOP.
P-NOTFND.
    MOVE 16 TO MESS-L.
    MOVE 'RECORD NOT FOUND' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PASK')  END-EXEC.
P-NOSTG.
    MOVE 10 TO MESS-L.
    MOVE 'NO STORAGE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    GO TO P-STOP.
P-ANYKEY.
    MOVE 32 TO MESS-L.
    MOVE 'DO NOT PRESS ENTER KEY TRY AGAIN' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT) LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PASK')  END-EXEC.
ERR-REC2.
    MOVE 'ACTIVE FROM STAT-FILE NOT FOUND' TO LERRO.
    GO TO P-SENDERR.
ERR-REC4.
    MOVE SPACE TO LERRO.
    MOVE 'RECORD ACTIVE FROM LOAN-FILE NOT FOUND' TO LERRO.
P-SENDERR.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
    EXEC CICS RETURN TRANSID ('PASK') END-EXEC.
ERR-REC2I.
    MOVE SPACE TO LERRO.
    MOVE 'PREVIOUS RECORD OF STAT FILE NOT FOUND' TO LERRO.
    GO TO P-SENDERR.
ERR-REC4I.
    MOVE SPACE TO LERRO.
    MOVE 'PREVIOUS RECORD OF LOAN FILE NOT FOUND' TO LERRO.
    GO TO P-SENDERR.
ERR-REC5.
    MOVE 'RECORD USE-ACTIVE FROM PERS-GUA-FILE NOT FOUND'
      TO LERRO.
ERR-REC6.
    MOVE 'RECORD ACTIVE FROM ASSET-GUA FILE NOT FOUND' TO LERRO.
    GO TO P-SENDERR.
ERR-REC7.
    MOVE 'RECORD USE-ACTIVE FROM LAND-STAT-FILE NOT FOUND'
        TO LERRO.
    GO TO P-SENDERR.
P-NOTLOAN.
    MOVE 'NOT ALLOW TO LOAN BECAUSE IT IS NOT DUE TIME TO LOAN'
        TO LERRO.
    GO TO P-SENDERR.
ERR-GUAM.
    MOVE 'ERROR RECORD ID-MEMBER NOT=ID-GUAED FROM PERS-GUA-FILE'
        TO LERRO.
    GO TO P-SENDERR.
ERR-GUAL.
    MOVE 'ERROR RECORD ID-MEMBER NOT=ID-GUAED FROM ASSET-GUA-FILE
-       '' TO LERRO.
    GO TO P-SENDERR.
P-ALLOW.
    MOVE SPACE TO LERRO.
    MOVE 'ALLOW TO GUAPANTY' TO LERRO.
```

```
    PERFORM P-MERR THRU P-MERR-EXIT.
P-ALLOW-EXIT. EXIT.
P-NOALLOW.
    MOVE SPACE TO LERRO.
    MOVE 'NOT ALLOW TO GUARANTY' TO LERRO.
    PERFORM P-MERR THRU P-MERR-EXIT.
P-NOALLOW-EXIT. EXIT.
P-NOLOAN.
    MOVE 'NOT ALLOW TO LOAN BECAUSE LOAN APPLY IS LESS THAN DEBT-
-       'REMAIN' TO LERRO.
    GO TO P-SENDERR.
P-NOGUA.
    MOVE 2 TO OTYPEO.
    MOVE ZEROES TO OGUA1O OAMTG1O OREM1O
    MOVE ZEROES TO OGUA2O OAMTG2O OREM2O
    MOVE ZEROES TO OGUA3O OAMTG3O OREM3O
    MOVE ZEROES TO OGUA4O OAMTG4O OREM4O
    MOVE ZEROES TO OGUA5O OAMTG5O OREM5O.
    MOVE 'HAVE NO GUARANTOR' TO LERRO.
    GO TO P-NEWLOAN.
P-STOPT.
    MOVE SPACE TO LERRO.
    MOVE 'THIS SCREEN IS ONLY INQUIRY' TO LERRO.
    GO TO P-SENDERR.
P-STOP.
    EXEC CICS SEND MAP ('LDESCT')
                  FROM (LDESCTO)  FREEKB
              END-EXEC.
    EXEC CICS RETURN TRANSID ('PASK') END-EXEC.
    STOP RUN.
```

ค.๔ โปรแกรมแสดงสถานภาพของสมาชิก (โปรแกรม P03)

```
IDENTIFICATION DIVISION.
PROGRAM-ID. PREPARE1.
AUTHOR. SAVING
DATE-COMPILED.
REMARKS. DISPLAY AND PREPARE TEMPORARY LOAN FILE.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
77  ID-MEM                  PIC  X(07)  VALUE SPACE.
77  MESS-L                  PIC  S9(4)  COMP VALUE +0.
77  MESS-OUT                PIC  X(75).
77  C-NINCOME               PIC  9(05)  VALUE 0.
77  C-TSHARE                PIC  9(06)  VALUE 0.
77  C-LRIGHT1               PIC  9(04)  VALUE 2500.
77  C-LPERIOD1              PIC  9(04)  VALUE 0.
77  C-PERIOD2               PIC  9(02)  VALUE 48.
77  C-LRIGHT2               PIC  9(05)  VALUE 0.
77  C-LPERIOD2              PIC  9(05)  VALUE 0.
77  C-PERIOD3               PIC  9(03)  VALUE 120.
77  C-LRIGHT3               PIC  9(06)  VALUE 300000.
77  C-LPERIOD3              PIC  9(05)  VALUE 0.
77  A-LTYPE                 PIC  X(01)  VALUE SPACE.
77  A-LAPPLY                PIC  9(06)  VALUE 0.
77  A-PERIOD                PIC  9(03)  VALUE 0.
77  A-LPERIOD               PIC  9(05)  VALUE 0.
77  A-LPERIOD1              PIC  9(05)  VALUE 0.
77  A-REMAIN                PIC  9(04)  VALUE 0.
77  I-RATE                  PIC  9(02)V9 VALUE 15.0.
77  A-INT                   PIC  9(04)V99 VALUE 0.
77  A-TOTALM                PIC  9(05)V99 VALUE 0.
77  EM-PRINC                PIC  9(05)  VALUE 0.
77  EM-DEBTRE               PIC  9(06)  VALUE 0.
77  EM-PERIOD               PIC  9(03)  VALUE 0.
77  OREX-PRINC              PIC  9(06)  VALUE 0.
77  CREX-DEBTRE             PIC  9(06)  VALUE 0.
77  CREX-PERIOD             PIC  9(03)  VALUE 0.
77  EM-CHECKL               PIC  X(03)  VALUE SPACE.
77  EM-LDATE                PIC  9(08)  VALUE 0.
77  CREX-CHECKL             PIC  X(03)  VALUE SPACE.
77  EM-TOTP                 PIC  9(03)  VALUE 0.
77  C-AMTREMAIN             PIC  9(05)  VALUE 0.
77  A-DEBTREMAIN            PIC  9(06)  VALUE 0.
77  A-INTDECRE              PIC  9(03)V99 VALUE 0.
77  K-TYPE                  PIC  9(01)  VALUE 0.
77  PDEBT                   PIC  X(01)  VALUE SPACE.
77  CHKS                    PIC  9(01)  VALUE 0.
77  CHKL                    PIC  9(01)  VALUE 0.
77  DATA-KEY                PIC  9(11)  VALUE 0.
77  NUMAPY                  PIC  9(04)  VALUE 0.

77  LR1                     PIC  9(06)  VALUE 0.
77  INT1                    PIC  9(05)V99  VALUE 0.
77  LP1                     PIC  9(05)  VALUE 0.
77  PR1                     PIC  9(03)  VALUE 0.
77  X1                      PIC  9(04)  VALUE 0.
77  X2                      PIC  9(04)  VALUE 0.
01  DFHAID      COPY DFHAID.
01  KEY-ID.
    02  ID-K                PIC  9(06).
    02  A-I                 PIC  X(01).
01  MEM-AGE.
    02  AGE-Y               PIC  Z9.
    02  FILLER              PIC  X(01)  VALUE '/'.
    02  AGE-M               PIC  Z9.
01  KEY-IDF4.
    02  IDF4                PIC  9(06).
    02  L-TYPE              PIC  X(01).
    02  A-I-L               PIC  X(01).
01  C-DATE.
    02  YY                  PIC  9(04).
    02  MM                  PIC  9(02).
    02  DD                  PIC  9(02).
01  P-DATE.
    02  DD1                 PIC  9(02).
    02  FILLER              PIC X(01)   VALUE '/'.
    02  MM1                 PIC  9(02).
    02  FILLER              PIC  X(01)  VALUE '/'.
    02  YY1                 PIC  9(04).
```

```
01  KEY-D.
    02  YY-D                    PIC  9(04).
    02  MM-D                    PIC  9(02).
    02  NO-D                    PIC  9(04).
    02  SE-D                    PIC  9(01).
01  MEMB-REC.
    02  ID-KEY-M.
        03  ID-MEMBER-M         PIC  9(06).
        03  A-I-M               PIC  X(01).
    02  CHECK-M.
        03  P-T-M               PIC  X(01).
        03  REC-STATUS-M        PIC  X(01).
        03  F-L-M               PIC  X(01).
    02  ID-MEMBER-CLD           PIC  9(04).
    02  NAME-THAI               PIC  X(60).
    02  NAME-ENG                PIC  X(40).
    02  SEX                     PIC  X(01).
    02  PAY-PLACE               PIC  9(03).
    02  ID-PAY                  PIC  X(07).
    02  ID-OFFICE               PIC  X(07).
    02  RE-MEMBER               PIC  X(01).
    02  NAME-COND               PIC  X(01).
    02  UPDATE-DATE-M           PIC  X(08).
    02  BLANK-M                 PIC  X(18).
01  STAT-REC.
    02  ID-KEY-S.
        03  ID-MEMBER-S         PIC  9(06).
        03  A-I-S               PIC  X(01).
    02  CHECK-S.
        03  P-T-S               PIC  X(01).
        03  REC-STATUS-S        PIC  X(01).
        03  F-L-S               PIC  X(01).
    02  MEMBER-DATE             PIC  9(08).
    02  MEMBER-AGE              PIC  9(03).
    02  MARITAL                 PIC  X(01).
    02  ID-SPOUSE               PIC  9(06).
    02  DEBT-CODE               PIC  X(03).
    02  OFFICE-CODE             PIC  X(03).
    02  MEMBER-TYPE             PIC  X(01).
    02  L-QUEUE                 PIC  9(02).
    02  AMT-LOAN                PIC  9(05).
    02  L-STATUS.
        03  EMERGENCY           PIC  X(01).
        03  CRDINARY            PIC  X(01).
        03  EXTRA               PIC  X(01).
    02  AMT-GUA                 PIC  9(05).
    02  GUA-LIMIT-USED          PIC  9(05).
    02  GUA-MEM-STATUS          PIC  9(01).
    02  GUA-MEM-RECORD          PIC  9(01).
    02  SALARY                  PIC  9(05).
    02  O-INCOME                PIC  9(04).
    02  XF-PRES-S               PIC  9(04).
    02  O-PAYMENT               PIC  9(05)V99.
    02  M-ADJUST                PIC  9(03)V99.
    02  OUT-COND                PIC  X(01).
    02  OUT-DATE                PIC  X(08).
    02  UPDATE-DATE-S           PIC  X(08).
    02  BLANK-S                 PIC  X(29).
01  XF-REC.
    02  ID-KEY-X.
        03  ID-MEMBER-X         PIC  9(06).
        03  A-I-X               PIC  X(01).
    02  CHECK-X.
        03  P-T-X               PIC  X(01).
        03  REC-STATUS-X        PIC  X(01).
        03  F-L-X               PIC  X(01).
    02  XF-RULE                 PIC  9(03).
    02  XF-PRES-X               PIC  9(04).
    02  XF-PREV                 PIC  9(04).
    02  XF-CHANGED-DATE         PIC  X(06).
    02  XF-COND                 PIC  X(01).
    02  XF-TIMES                PIC  9(01).
    02  TOTAL-SHARE-COND        PIC  X(01).
    02  TOTAL-SHARE             PIC  9(06).
```

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

```
           02  DIVIDEND-CAL                PIC  9(07).
           02  UPDATE-DATE-X               PIC  X(08).
           02  BLANK-X                     PIC  X(13).
       01  LOAN-REC.
           02  ID-KEY-L.
               03  ID-MEMBER-L             PIC  9(06).
               03  L-TYPE-L                PIC  X(01).
               03  A-I-L1                  PIC  X(01).
           02  CHECK-L.
               03  P-T-L1                  PIC  X(01).
               03  REC-STATUS-L            PIC  X(01).
               03  F-L-L                   PIC  X(01).
           02  L-AGREE-NO                  PIC  9(07).
           02  L-DATE                      PIC  9(08).
           02  PRINCIPAL-L                 PIC  9(06).
           02  L-FLAG                      PIC  X(01).
           02  PRINC-F-L-PAID              PIC  9(05).
           02  PRINC-MONTHLY-PAID          PIC  9(05).
           02  DEBT-REMAIN                 PIC  9(06).
           02  TOTAL-PERIOD                PIC  9(03).
           02  PERIOD                      PIC  9(03).
           02  INT-F-L-PAID                PIC  9(04)V99.
           02  INT-PRES-PAID               PIC  9(04)V99.
           02  INT-DECRE                   PIC  9(03)V99.
           02  INT-RATE                    PIC  9(02)V9.
           02  INTX                        PIC  9(05)V99.
           02  GUA-MEN-KEY-L.
               03  ID-GUA-MEN-L            PIC  9(06).
               03  SECVENCE-L2             PIC  9(01).
           02  GUA-LAND-KEY-L.
               03  ID-GUA-LAND-L           PIC  9(09).
               03  SECVENCE-L3             PIC  9(01).
           02  GUA-LAND-NUMB               PIC  9(01).
           02  GUA-SHARE                   PIC  9(06).
           02  GUA-DEPOSIT                 PIC  9(06).
           02  UPDATE-DATE-L               PIC  X(08).
           02  BLANK-L                     PIC  X(08).
       01  TEMP-LOAN-REC.
           02  ID-KEY-T.
               03  ID-MEMBER-T             PIC  9(06).
               03  L-TYPE-T                PIC  X(01).
               03  A-I-T1                  PIC  X(01).
           02  CHECK-T.
               03  P-T-T1                  PIC  X(01).
               03  REC-STATUS-T            PIC  X(01).
               03  F-L-T                   PIC  X(01).
           02  L-AGREE-NO-T                PIC  9(07).
           02  L-DATE-T                    PIC  9(08).
           02  PRINCIPAL-T                 PIC  9(06).
           02  L-FLAG-T                    PIC  X(01).
           02  PRINC-F-L-PAID-T            PIC  9(05).
           02  PRINC-MONTHLY-PAID-T        PIC  9(05).
           02  DEBT-REMAIN-T               PIC  9(06).
           02  TOTAL-PERIOD-T              PIC  9(03).
           02  PERIOD-T                    PIC  9(03).
           02  INT-F-L-PAID-T              PIC  9(04)V99.
           02  INT-PRES-PAID               PIC  9(04)V99.
           02  INT-DECRE-T                 PIC  9(03)V99.
           02  INT-RATE-T                  PIC  9(02)V9.
           02  INTX-T                      PIC  9(05)V99.
           02  GUA-MEN-KEY-T.
               03  ID-GUA-MEN-T            PIC  9(06).
               03  SEQVENCE-T2             PIC  9(01).
           02  GUA-LAND-KEY-T.
               03  ID-GUA-LAND-T           PIC  9(09).
               03  SECUENCE-T3             PIC  9(01).
           02  GUA-LAND-NUMB-T             PIC  9(01).
           02  GUA-SHARE-T                 PIC  9(06).
           02  GUA-DEPOSIT-T               PIC  9(06).
           02  UPDATE-DATE-T               PIC  X(08).
           02  BLANK-T                     PIC  X(08).
       01  DATA-REC.
           02  ID-TRAN-KEY.
               03  YEAR-TR                 PIC  9(04).
               03  MONTH-TR                PIC  9(02).
               03  APPLY-NO                PIC  9(04).
               03  SEQUENCE-NO             PIC  9(01).
```

```
           02  IDTR                    PIC  9(06).
           02  L-TYPE-TR               PIC  X(01).
           02  PRINCIPAL-TR            PIC  9(06).
           02  TOTAL-PERIOD-TR         PIC  9(03).
           02  ID-GUA-TR               PIC  9(09).
           02  GUA-STAT-TR             PIC  9(01).
           02  GUA-NUM-TR              PIC  9(01).
           02  GUA-LIMIT-TR            PIC  9(06).
           02  GUA-SHARE-TR            PIC  9(06).
           02  GUA-DEPOSIT-TR          PIC  9(06).
           02  EVAL-DATE-TR            PIC  9(08).
           02  LAND-AREA-TR            PIC  9(04).
           02  EVAL-UNIT-TR            PIC  9(06).
           02  EVAL-BUILDING-TR        PIC  9(07).
           02  EVAL-INSURANCE-TR       PIC  9(06).
           02  UPDATE-DATE-TR          PIC  9(08).
           02  BLANK-TR                PIC  X(33).
       01  HOME-REC                    PIC  X(128).
       01  TEMP-WORK.
           02  TEMP-KEY                PIC  X(08).
           02  TEMP-DATA               PIC  X(120).
       01  IMAPI       COPY IMAP.
       01  CLOSTATI    COPY OLOSTAT.
       01  MAPPLY1I    COPY MAPPLY1.
       01  MWRITEI     COPY MWRITE.
       01  MAPERR1I    COPY  MAPERR1.
       PROCEDURE DIVISION.
       P-START.
           EXEC CICS HANDLE CONDITION
                     INVMPSZ (P-INVMPSZ)
                     MAPFAIL (P-MAPFAIL)
                     DSIDERR (P-DSIDERR)
                     LENGERR (P-LENGERR)
                     NOTFND  (P-NOTFND)
                     NOSTG (P-NOSTG)  END-EXEC.
           EXEC CICS HANDLE AID
                     CLEAR (P-CLEAR)
                     ANYKEY  (P-ANYKEY) END-EXEC.
           MOVE 90000000000 TO KEY-D.
           EXEC CICS READ INTO (DATA-REC)
                          DATASET ('CIMAFL')
                          RIDFLD (KEY-D)
                          KEYLENGTH (11) END-EXEC.
           MOVE EVAL-DATE-TR TO C-DATE.
           MOVE YY TO YY1.
           MOVE MM TO MM1.
           MOVE DD TO DD1.
           MOVE P-DATE TO KDATEO.
       P-INPUT.
           MOVE LOW-VALUE TO CLOSTATO.
           MOVE LOW-VALUE TO DDEBTO.
           MOVE LOW-VALUE TO MAPERR1O.
           EXEC CICS SEND MAP('IMAP')
                          MAPONLY
                          FREEKB ERASE
                          END-EXEC.
           EXEC CICS RECEIVE MAP ('IMAP') INTO (IMAPI) END-EXEC.
           MOVE KEYLNI TO ID-K.
           MOVE 0 TO A-I.
           MOVE KEY-ID TO ID-MEM.
           EXEC CICS READ INTO (MEMB-REC)
                          DATASET ('MEMBFL')
                          RIDFLD (ID-MEM)
                          KEYLENGTH (7)  END-EXEC.
           MOVE NAME-ENG TO KNAMEC NAME1O.
           EXEC CICS SEND MAP ('IMAP')
                          FROM (IMAPO)
                          END-EXEC.
           MOVE P-DATE TO PDATEO.
           MOVE ID-K TO KEYIDMEO.
       P-READF2.
           EXEC CICS READ INTO (STAT-REC)
                          DATASET ('STATFL')
                          RIDFLD (ID-MEM)
                          KEYLENGTH (7)  END-EXEC.
       P-READF2-EXIT. EXIT.
```

```
P-CHECKCODE1.
    IF CHECK-S = '000' GO TO P-SENDM1.
    IF CHECK-S NOT = '110' MOVE 'RECORD ACTIVE NOT FOUND'
                                        TO LERRO
                                    GO TO P-SENDMERR.
    MOVE '1' TO A-I.
    MOVE KEY-ID TO ID-MEM.
    PERFORM P-READF2 THRU P-READF2-EXIT.
    IF CHECK-S NOT = '100'
            MOVE 'RECORD OLD TEMPORARY TO BE DELETED NOT FOUND'
                TO LERRO
            GO TO P-SENDMERR.
    MOVE 'C' TO A-I.
    MOVE KEY-ID TO ID-MEM.
    PERFORM P-READF2 THRU P-READF2-EXIT.
P-SENDM1.
    MOVE MEMBER-TYPE TO MEMTYPEO.
    DIVIDE MEMBER-AGE BY 12 GIVING AGE-Y REMAINDER AGE-M.
    MOVE MEM-AGE TO MEMAGEO.
    MOVE SALARY TO INCOMEO.
    COMPUTE C-NINCOME = (SALARY + C-INCOME) - (C-PAYMENT - M-ADJU
    ST).
    MOVE C-NINCOME TO NINCOMEO.
P-READF3.
    MOVE '0' TO A-I.
    MOVE KEY-ID TO ID-MEM.
    EXEC CICS READ INTO (XF-REC)
                DATASET ('XFFL')
                RIDFLD (ID-MEM)
                KEYLENGTH (7) END-EXEC.
P-READF3-EXIT.  EXIT.
P-CHECKCODE2.
    IF CHECK-X = '000' GO TO P-CHECKDATE.
    IF CHECK-X NOT = '110'
        MOVE 'RECORD ACTIVE NOT FOUND' TO LERRO
        GO TO P-SENDMERR.
    MOVE '1' TO A-I.
    MOVE KEY-ID TO ID-MEM.
    PERFORM P-READF3 THRU P-READF3-EXIT.
    IF CHECK-X NOT = '100'
        MOVE 'RECORD OLD TEMPOLARY TO BE DELETED NOT FOUND'
            TO LERRO
        GO TO P-SENDMERR.
    MOVE '0' TO A-I.
    MOVE KEY-ID TO ID-MEM.
    PERFORM P-READF3 THRU P-READF3-EXIT.
P-CHECKDATE.
    IF DD > 10
        MOVE TOTAL-SHARE TO TOSHAREO
    ELSE COMPUTE C-TSHARE = (TOTAL-SHARE * XF-PRES-X)
        MOVE C-TSHARE TO TOSHAREO.
    MOVE XF-PRES-X TO XFPRESO.
*****COMPUTE LEAN RIGHT*****
P-SRIGHT1.
    COMPUTE LR1 = (C-NINCOME * 2 * 1200) / (1200 + (I-RATE * 2)).
    IF LR1 > C-LRIGHT1
        COMPUTE LR1 = C-LRIGHT1
        COMPUTE C-LPERIOD1 = (C-LRIGHT1 / 2)
        COMPUTE INT1 = (C-LRIGHT1 * I-RATE) / 1200
        GO TO P-M1.
    COMPUTE LR1 = (LR1 / 20 + 1) * 20.
    COMPUTE C-LPERIOD1 = (LR1 / 2).
    COMPUTE INT1 = (LR1 * I-RATE) / 1200.
P-M1.
    MOVE '1' TO LOTYPE10.
    MOVE LR1 TO LORATE10.
    MOVE C-LPERIOD1 TO LPERIOD10.
    MOVE INT1 TO INTEO.
    MOVE 2 TO PERIOD10.
P-SRIGHT2.
    MOVE ZERO TO LR1 LP1 PR1 INT1.
    COMPUTE INT1 = (AMT-GUA * I-RATE) / 1200.
    COMPUTE LP1 = C-NINCOME - INT1.
    DIVIDE LP1 BY 20 GIVING X1 REMAINDER X2.
    IF X2 = 0 NEXT SENTENCE
    ELSE COMPUTE LP1 = (X1 + 1) * 20.
    COMPUTE PR1 = AMT-GUA / LP1.
```

```
           COMPUTE LR1 = LP1 * PR1.
           COMPUTE INT1 = (LR1 * I-RATE1 / 1200.
           MOVE '2' TO LCTYPE2O.
           MOVE LR1 TO LORATE2O.
           MOVE LP1 TO LPERIO2O.
           MOVE PR1 TO PERIOD2O.
           MOVE INT1 TO INTORC.
       P-SKIGHT3.
           MOVE ZERO TO LR1 LP1 PR1 INT1.
           MOVE ZERO TO X1 X2.
           COMPUTE LR1 = (C-NINCOME * 12CC) / (I-RATE * 10).
           IF LR1-> C-LRIGHT3
               COMPUTE LR1 = C-LRIGHT3.
           COMPUTE INT1 = (LR1 * I-RATE) / 1200.
           COMPUTE LP1 = LR1 / C-PERIOC3.
           DIVIDE LP1 BY 20 GIVING X1 REMAINDER X2.
           IF X2 = 0 NEXT SENTENCE
           ELSE COMPUTE LP1 = (X1 + 1) * 20.
           COMPUTE PR1 = C-PERICD3.
           COMPUTE LR1 = LP1 * PR1.
           COMPUTE INT1 = (LR1 * I-RATE) / 1200.
           MOVE '3' TO LCTYPE3O.
           MOVE LR1 TO LORATE2O.
           MOVE LP1 TO LPERIO3O.
           MOVE INT1 TO INTEXO.
       *****SEND MAP LOAN-STATUS1****
           EXEC CICS SEND MAP ('CLOSTAT')
                          FROM (OLGSTATO) ERASE
                     END-EXEC.
       *****CONSIDERATION OLD DEBT****
       READ-F2AGAIN.
           MOVE O TO CHKS CHKL.
           MOVE SPACE TO PDEBT.
           MOVE '0' TO A-I.
           MOVE KEY-ID TO ID-MEM
           PERFORM P-READF2 THRU P-READF2-EXIT.
           IF CHECK-S = '000' OR '110' GO TO CHECK-LSTATUS.
           MOVE 'RECORD ACTIVE NOT FOUND' TO LERRO.
           GO TO P-SENDMERR.
           MOVE 0 TO EM-PRINC EM-DEBTRE EM-PERIOD
                     EM-LDATE EM-TOTP.
       CHECK-LSTATUS.
           MOVE ALL '=' TO LINE1C.
           IF L-STATUS = '000'
               MOVE 0 TO OLDL1O OLDL2O EM-CHECKL OREX-CHECKL
               MOVE 0 TO PRINCE1O DEBTRE1C FPERID1O OREX-PRINC
                         PRINCE2O DEBTRE2O FPERID2O OREX-DEBTRE
                         CREX-PERIOD
                         EM-PRINC EM-DEBTRE EM-PERIOD EM-LDATE EM-TOTP
               MOVE SPACE TO TORP1O TORP2O
           PERFORM P-NOLOAN THRU P-NOLOAN-EXIT
               GO TO P-RECEIVEP.
           IF EMERGENCY = '1' GO TO P-CDEBT1.
           MOVE 0 TO OLDL1O EM-CHECKL.
           MOVE ZERO TO PRINCE1C DEBTRE1O FPERID1C.
           MOVE SPACE TO TORP1O.
       CHECK-LSTAT2.
           IF ORDINARY = '2' GO TO P-CDEBT2.
       *****HAVE EXTRA LOAN****
           MOVE IO-K TO IDF4.
           MOVE EXTRA TO L-TYPE.
           IF CHKS = 1 MOVE '1' TO A-I-L
           ELSE MOVE '0' TO A-I-L.
       P-READF4.
           EXEC CICS READ INTO (LOAN-REC)
                        DATASET ('LOANFL')
                        RIDFLD (KEY-IDF4)
                        KEYLENGTH (8) END-EXEC.
       P-READF4-EXIT. .EXIT.
       P-CHECKL.
           IF CHKS = 1 AND CHECK-L = '100' MOVE '(PREVIOUS)' TO TORP2O
                                           GO TO P-CDEBT3.
           IF CHECK-L = '000' MOVE '(PERMANENT)' TO TORP2O
                              GO TO P-CDEBT3.
           IF CHECK-L = '110' MOVE '(TEMPORARY)' TO TORP2O
```

```
           ELSE
               MOVE 'RECORD ACTIVE NOT FOUND' TO LERRC
               GO TO P-SENDMERR.
       P-CDEBT3.
           MOVE L-TYPE-L TO CLDL2C.
           MOVE PRINCIPAL-L TO PRINCE2C CREX-PRINC.
           MOVE DEBT-REMAIN TO DEBTRE2C CREX-DEBTRE.
           MOVE PERICD TC FPERID2C CREX-PERICD.
       P-RECEIVEP.
           MOVE SPACE TO KDEBTC.
           EXEC CICS SEND MAP ('CDEBT')
                          MAPSET ('CLCSTAT')
                          FROM (CDEBTC)
                          END-EXEC.
           EXEC CICS RECEIVE MAP ('CDEBT')
                             INTC (CDEBTI)
                             MAPSET ('CLCSTAT') END-EXEC.
           MOVE KDEBTI TO PDEBT.
           IF PDEBT = 'Y' GO TO P-KPRE.
           IF CHKS = 1 GO TO READ-F2AGAIN.
           GO TO P-MAPCAL.
       *****READ PREVIOUS STAT-FILE RECCRD****
       P-KPRE.
           IF CHECK-S = '000' GO TO P-MAPCAL.
           MOVE '1' TC A-I.
           MOVE KEY-ID TO ID-MEM.
           PERFCRM P-READF2 THRU P-READF2-EXIT.
           IF CHECK-S = '100' MOVE 1 TC CHKS
                              GC TO CHECK-LSTATUS.
           MOVE SPACE TC LERRC.
           MOVE 'PREVICUS RECCRD CF STAT-FILE NOT FCUND' TO LERRC.
           GO TC P-SENDMERR.
       P-CDEBT1.
       *****HAVE EMERGENCY LCAN****
           MOVE ID-K TO IDF4.
           MOVE EMERGENCY TO L-TYPE.
           IF CHKL = 1 MOVE 0 TO A-I-L
                       PERFCRM P-READF4 THRU P-READF4-EXIT
                       GO TO P-CCN1.
           IF CHKS = 1 MOVE 1 TC A-I-L
           ELSE  MOVE 0 TC A-I-L.
           PERFORM P-READF4 THRU P-READF4-EXIT.
       ****CCNSIDER EMERGENCY****
       P-CCN1.
           IF CHKS = 1 AND CHECK-L = '000' OR '100'
               MOVE '(PREVICUS)' TO TORP1C
               GO TO P-CDEBT11.
           IF CHECK-L = '000' MOVE 1 TC CHKL
                              MCVE '(PERMANENT)' TC TCRP1C
                              GO TO P-CDEBT11.
           IF CHECK-L = '110' MOVE '(TEMPCRARY)' TO TORP1C
           ELSE
               MOVE 'RECORD ACTIVE NCT FCUND' TO LERRC
               GO TO P-SENDMERR.
       P-CDEBT11.
           MOVE L-TYPE-L TO CLDL1C.
           MOVE PRINCIPAL-L TO PRINCE1C EM-PRINC.
           MOVE DEBT-REMAIN TC DEBTRE1C EM-DEBTRE.
           MOVE PERICD TC FPERID1C EM-PERICD.
           MOVE CHECK-L TO EM-CHECKL.
           MOVE L-CATE TO EM-LDATE.
           MOVE TCTAL-PERIOD TO EM-TOTP.
           IF ORDINARY = '2' GO TC P-CDEBT2.
           MOVE SPACE TC CLDL2C CREX-CHECKL
           MOVE SPACE TO TORP2C.
           MOVE 0 TC PRINCE2C DEBTRE2C FPERID2C
           MOVE 0 TO CREX-PRINC CREX-DEBTRE CREX-PERICD
           IF PDEBT = 'Y' PERFORM P-MAPCDEBT THRU P-MAPCDEBT-EXIT
                          GC TC READ-F2AGAIN.
           GO TO P-RECEIVEP.
       P-CDEBT2.
       *****HAVE ORDINARY LCAN***
           MOVE ID-K TO IDF4.
           MOVE ORDINARY TO L-TYPE.
           IF CHKL = 1 MOVE 0 TC A-I-L
                       GO TC P-READF4.
           IF CHKS = 1 MOVE 1 TC A-I-L
           ELSE MOVE C TO A-I-L.
           GC TC P-READF4.
```

```
****SEND MAP OLD DEBT****
P-MAPODEBT.
    MOVE SPACE TO LERRO.
    EXEC CICS SEND MAP ('ODEBT')
                  MAPSET ('OLOSTAT')
                  FROM (ODEBTO)
                  END-EXEC.
P-MAPODEBT-EXIT. EXIT.
*****SEND MAP CALCULATION****
P-MAPCAL.
    MOVE LOW-VALUE TO MAPPLY1O MAPERR1O.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
    MOVE LOW-VALUE TO MWRITEO.
    EXEC CICS SEND MAP ('MAPPLY1')
                  FROM (MAPPLY1O)
                  END-EXEC.
    EXEC CICS RECEIVE MAP ('MAPPLY1')
                  INTO (MAPPLY1I)
                  END-EXEC.
P-MAPCAL-EXIT. EXIT.
P-MOVEI.
    MOVE LOTYPEAI TO A-LTYPE.
    IF A-LTYPE = '5' GO TO P-STOP.
    MOVE LOAPPLY1 TO A-LAPPLY.
    MOVE PERIOD1I TO A-PERIOD.
    IF A-LTYPE = '1' GO TO P-CHECKL1.
    IF A-LTYPE = '2' GO TO P-CHECKR2.
    IF A-LTYPE = '3' OR '4' GO TO P-CHECKR3.
    MOVE SPACE TO LERRO.
    MOVE 'LOAN-TYPE CODE IS WRONG' TO LERRO.
    GO TO P-SENDMERR.
P-CHECKL1.
    IF EMERGENCY = '1' GO TO P-CHECKTEM1.
    IF ORDINARY  = '2' GO TO P-CHECKL2.
    IF EXTRA = '3' OR '4' GO TO P-EXLOAN.
P-CHECKR1.
    MOVE 0 TO A-LPERIOD1 A-REMAIN A-LPERIOD.
    IF A-LAPPLY > C-LRIGHT1 GO TO P-OPTION.
    DIVIDE A-LAPPLY BY A-PERIOD GIVING A-LPERIOD REMAINDER
    A-REMAIN.
    COMPUTE A-LPERIOD1 = A-LPERIOD + A-REMAIN.
    IF OD > 10 COMPUTE A-INT = (A-LPERIOD * 2 * I-RATE) / 1200
              GO TO P-COMTOTAL.
    COMPUTE A-INT = (A-LAPPLY * I-RATE) / 1200.
P-COMTOTAL.
    COMPUTE A-TOTALM = (A-LPERIOD1 + A-INT)
    MOVE A-LAPPLY TO LOAPPLYO.
    MOVE A-LPERIOD TO APERIDIO.
    MOVE A-LPERIOD1 TO LPERIOD.
    MOVE A-INT TO LINTAO.
    MOVE A-TOTALM TO TOTALMO.
*****SEND MAP APPLY AGAIN****
A-AGAIN.
    MOVE SPACE TO LERRO.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
    EXEC CICS SEND MAP ('MAPPLY1')
                  FROM (MAPPLY1O) END-EXEC.
A-EXIT. EXIT.
P-IS.
    GO TO P-CHOSEN.
P-OPTION.
    MOVE SPACE TO LERRO.
    MOVE 'LOAN APPLY IS MORE THAN LOAN RIGHT' TO LERRO.
P-ERR.
    EXEC CICS SEND MAP ('MAPERR1')
                  FROM (MAPERR1O)  END-EXEC.
    GO TO P-MAPCAL.
P-CHECKTEM1.
    MOVE SPACE TO LERRO.
    MOVE 'NOT ALLOW TO LOAN BECAUSE OF HAVE LOANED' TO LERRO.
    GO TO P-SENDMERR.
P-CHECKL2.
    IF OREX-PRINC NOT < AMT-LOAN
       MOVE SPACE TO LERRO
```

```
           MOVE 'NOT ALLOW TO LOAN BECAUSE OF HAVE LOANED LOCAL AMT
-          '-LOAN' TO LERRO
           GO TO P-SENDMERR.
       COMPUTE C-AMTREMAIN = (AMT-LOAN - OREX-PRINC).
       IF A-LAPPLY NOT > C-AMTREMAIN GO TO P-CHECKR1.
       GO TO P-SENDPRE.
   P-CHECKR2.
       IF L-QUEUE = 0 NEXT SENTENCE
       ELSE GO TO P-NORIGHTL.
       IF A-LAPPLY < OREX-DEBTRE GO TO P-SENDPRE.
       COMPUTE C-AMTREMAIN = (AMT-LOAN - EM-PRINC).
       IF A-LAPPLY NOT > C-AMTREMAIN GO TO P-CALPERIOD.
       MOVE SPACE TO LERRC.
       MOVE 'LOAN APPLY IS MORE THAN AMT-LOAN REMAIN' TO LERRC.
       GO TO P-ERR.
   P-CALPERIOD.
       MOVE 0 TO A-LPERIOD A-LPERIOD1 A-REMAIN.
       DIVIDE A-LAPPLY BY A-PERIOD GIVING A-LPERIOD REMAINDER
           A-REMAIN.
       COMPUTE A-LPERIOD1 = A-LPERIOD + A-REMAIN.
       IF A-LPERIOD1 > C-NINCOME
           GO TO P-PRESSCHANGE2.
       COMPUTE A-INT = (A-LAPPLY * I-RATE) / 1200.
       COMPUTE A-TOTALM = (A-LPERIOD + A-INT).
       IF A-TOTALM > C-NINCOME
           MOVE SPACE TO LERRC
           MOVE '**FIRST PAYMENT IS MORE THAN NET INCOME**' TO
               LERRC
           GO TO P-ERR.
       MOVE A-LTYPE TO LOTYPEAC.
       MOVE A-LAPPLY TO LOAPPLYO.
       MOVE A-PERIOD TO PERIODLO.
       MOVE A-LPERIOD TO APERIODC.
       MOVE A-LPERIOD1 TO LPERIOC.
       MOVE A-INT TO LINTAO.
       MOVE A-TOTALM TO TOTALMO.
       PERFORM A-AGAIN THRU A-EXIT.
   P-CHOSEW.
       MOVE SPACE TO OPTION2C.
       EXEC CICS SEND MAP ('MWRITE')
                       FROM (MWRITEO)  END-EXEC.
       EXEC CICS RECEIVE MAP ('MWRITE') INTO (MWRITEI)
                       END-EXEC.
       IF OPTION2I = 'Y' MOVE NOAPYI TO NUMAPY
                         GO TO P-USE.
       GO TO P-MAPCAL.
   P-CHECKR3.
       IF L-QUEUE = 0 NEXT SENTENCE
       ELSE GO TO P-NORIGHTL.
       IF A-LAPPLY < OREX-DEBTRE GO TO P-SENDPRE.
       IF A-LAPPLY > C-LRIGHT3 GO TO P-OPTION.
       GO TO P-CALPERIOD.
   *****MOVE-STORY-TO-TEMP RECORD***
   P-USE.
       EXEC CICS HANDLE CONDITION
                   NOTFND (P-C4) END-EXEC.
       MOVE ID-K TO IDF4.
       ADD 1 TO K-TYPE.
       MOVE K-TYPE TO L-TYPE.
       MOVE '0' TO A-I-L.
       EXEC CICS READ INTO (TEMP-LOAN-REC)
                   DATASET ('TEMPFL')
                   RIDFLD (KEY-IDF4)
                   KEYLENGTH (8) END-EXEC.
       IF CHECK-T = 'ILO' GO TO P-CHECKLOANTYPE.
   P-C4.
       IF K-TYPE < 4 GO TO P-USE.
       GO TO P-MOVETEMP.
   P-CHECKLOANTYPE.
       IF K-TYPE = 1
           MOVE '$$NOT ALLOW TO LOAN BECAUSE HAVE PREPARED EMERGENCY
-          ' LOAN' TO LERRO
           GO TO P-SENDMERR.
       IF K-TYPE = 2
           MOVE '**NOT ALLOW TO LOAN BECALSE HAVE PREPARED ORDINARY
-          'LOAN' TO LERRC
           GO TO P-SENDMERR.
       MOVE '**NOT ALLOW TO LOAN BECALSE HAVE PREPARED EXTRA LOAN'
       TO LERRO.
       GO TO P-SENDMERR.
```

```
P-MOVETEMP.
    MOVE ZERO TO DATA-REC.
    MOVE ID-K TO ID-MEMBER-T IDTR.
    MOVE A-LTYPE TO L-TYPE-T L-TYPE-TR.
    MOVE '0' TO A-I-T1.
    MOVE '11C' TO CHECK-T.
    MOVE 0 TO L-AGREE-NO-T.
    MOVE C-DATE TO L-DATE-T.
    MOVE A-LAPPLY TO PRINCIPAL-T PRINCIPAL-TR.
    MOVE '1' TO L-FLAG-T.
    MOVE A-LPERIOD1 TO PRINC-F-L-PAID-T.
    MOVE A-LPERIOD TO PRINC-MONTHLY-PAID-T.
    COMPUTE A-DEBTREMAIN = (A-LAPPLY - A-LPERIOD1).
    MOVE A-DEBTREMAIN TO DEBT-REMAIN-T.
    MOVE A-PERIOD TO TOTAL-PERIOD-T TOTAL-PERIOD-TR.
    MOVE 1 TO PERIOD-T.
    MOVE A-INT TO INT-F-L-PAID-T.
    COMPUTE A-INTDECRE = (A-LPERIOD * I-RATE) / 1200.
    MOVE A-INTDECRE TO INT-DECRE-T.
    MOVE I-RATE TO INT-RATE-T.
    MOVE 0 TO INTX-T.
    MOVE ZERO TO GUA-PEN-KEY-T GUA-LAND-KEY-T GUA-LAND-NUMB-T
         GUA-SHARE-T GUA-DEPOSIT-T.
    MOVE C-DATE TO UPDATE-DATE-T UPDATE-DATE-TR.
    MOVE DATA-REC TO HOME-REC.
*****WRITE RECORD IN TEMPORARY LOAN FILE*****
    EXEC CICS HANDLE CONDITION
              NOTFND (P-WRITE)
                     DUPREC (P-DUPREC) END-EXEC.
    MOVE ID-K TO IDF4.
    MOVE A-LTYPE TO L-TYPE.
    MOVE 'C' TO A-I-L.
P-READ8.
    MOVE KEY-IDF4 TO TEMP-KEY.
    EXEC CICS READ INTO (TEMP-WORK)
              DATASET ('TEMPFL')
              RIDFLD (KEY-IDF4)
              UPDATE
              KEYLENGTH (8) END-EXEC.
    IF CHECK-T = '000' OR '110' GO TO P-NOADD.
P-REWRITE.
    EXEC CICS REWRITE DATASET ('TEMPFL')
              FROM (TEMP-LOAN-REC)
                   END-EXEC.
P-CARD.
    MOVE YY TO YY-D.
    MOVE MM TO MM-D.
    MOVE NUMAPY TO NO-D.
    MOVE 1 TO SE-D.
    EXEC CICS HANDLE CONDITION
                   NOTFND (P-WR) END-EXEC.
    EXEC CICS READ INTO (DATA-REC)
                   DATASET ('CIMAFL')
                   RIDFLD (KEY-D)
                   UPDATE
                   KEYLENGTH (11) END-EXEC.
    MOVE HOME-REC TO DATA-REC.
    MOVE KEY-D TO ID-TRAN-KEY.
    EXEC CICS REWRITE DATASET ('CIMAFL')
                   FROM (DATA-REC) END-EXEC.
    PERFORM P-READY THRU P-READY-EXIT.
    EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-WR.
    MOVE HOME-REC TO DATA-REC.
    MOVE KEY-D TO ID-TRAN-KEY.
    EXEC CICS WRITE FROM (DATA-REC)
                   DATASET ('CIMAFL')
                   RIDFLD (KEY-D)
                   KEYLENGTH (11) END-EXEC.
    PERFORM P-READY THRU P-READY-EXIT.
    EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-WRITE.
    EXEC CICS WRITE FROM (TEMP-LOAN-REC)
```

```
                    DATASET ('TEMPFL')
                    RIDFLD (KEY-IDF4)
                    KEYLENGTH (8) END-EXEC.
     GO TO P-CARD.
P-NOADD.
     MOVE SPACE TO LERRC.
     MOVE '**DO NOT ADD RECORD BECAUSE OF USEFUL RECORD' TO LERRO.
     GO TO P-SENDMERR.
P-READY.
     MOVE SPACE TO LERRC.
     MOVE '***PREPARED CARD IMAGE ALREADY***' TO LERRC.
     EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
P-READY-EXIT. EXIT.
P-DUPREC.
     MOVE 36 TO MESS-L.
     MOVE 'WRITE ERROR BECAUSE DUPLICATE RECORD' TO MESS-OUT.
     EXEC CICS SEND TEXT FROM (MESS-OUT)
                    LENGTH (MESS-L)
                    ERASE
                    WAIT END-EXEC.
     EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-SENDTEMP.
     MOVE SPACE TO LERRC.
     MOVE 'NOT ALLOW TO LOAN BECAUSE OF TEMPORARY RECORD'
         TO LERRC.
     GO TO P-SENDMERR.
P-EMLOAN.
     MOVE SPACE TO LERRC.
     MOVE 'NOT ALLOW TO LOAN BECAUSE HAVE EMERGENCY LOAN'
         TO LERRC.
     GO TO P-SENDMERR.
P-EXLOAN.
     MOVE SPACE TO LERRO.
     MOVE 'NOT ALLOW TO LOAN BECAUSE HAVE EXTRA LOAN' TO LERRO.
     GO TO P-SENDMERR.
P-NORIGHTL.
     MOVE SPACE TO LERRO.
     MOVE 'NOT ALLOW TO LOAN BECAUSE IT IS NOT DUE TIME TO LOAN'
         TO LERRC.
     GO TO P-SENDMERR.
P-SENDPRE.
     MOVE SPACE TO LERRO.
     MOVE 'NOT ALLOW TO LOAN BECAUSE LOAN APPLY IS LESS THAN DEBT-
          'REMAIN' TO LERRO.
     GO TO P-ERR.
P-PRESSCHANGE2.
     MOVE SPACE TO LERRO.
     MOVE 'PRINCIPAL-FIRST-PAID IS MORE THAN NET-INCOME, PLEASE
          'CHANGE PERICD' TO LERRC.
     GO TO P-ERR.
P-PRESSCHANGE2-EXIT.  EXIT.
P-STOP.
     MOVE SPACE TO LERRO.
     MOVE 'THIS SCREEN IS ONLY INQUIRY' TO LERRO.
P-SENDMERR.
     EXEC CICS SEND MAP ('MAPERR1')
                    FROM (MAPERR1O)
                    END-EXEC.
     EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-SENDMERR-EXIT.  EXIT.
P-CLEAR.
     EXEC CICS RETURN END-EXEC.
P-NOTFND.
     MOVE 16 TO MESS-L.
     MOVE 'RECORD NOT FOUND' TO MESS-OUT.
     EXEC CICS SEND TEXT FROM (MESS-OUT)
                    LENGTH (MESS-L)
                    WAIT  END-EXEC.
     EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-INVMPSZ.
     MOVE 16 TO MESS-L.
     MOVE 'INVALID MAP SIZE' TO MESS-OUT.
     EXEC CICS SEND TEXT FROM (MESS-OUT)
                    LENGTH (MESS-L)
                    WAIT END-EXEC.
     EXEC CICS RETURN END-EXEC.
```

```
P-MAPFAIL.
    MOVE 8 TO MESS-L.
    MOVE 'MAP FAIL' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-DSIDERR.
    MOVE 17 TO MESS-L.
    MOVE 'DATA SET ID ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-LENGERR.
    MOVE 18 TO MESS-L.
    MOVE 'WRONG LENGTH ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-NOSTG.
    MOVE 10 TO MESS-L.
    MOVE 'NO STORAGE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-ANYKEY.
    MOVE 32 TO MESS-L.
    MOVE 'OOPS NOT PRESS ENTER KEY TRY AGAIN' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('PREP') END-EXEC.
P-NOLOAN.
    MOVE SPACE TO LERRO.
    MOVE '***FREE DEBT***' TO LERRO.
    EXEC CICS SEND MAP ('MAPERR1')
                   FROM (MAPERR1O)
                   END-EXEC.
P-NOLOAN-EXIT. EXIT.
END-RTN.
    STOP RUN.
```

ค.๔ โปรแกรมแสดงความสามารถในการค้ำประกันของสมาชิก (โปรแกรม P04)

```
ID DIVISION.
PROGRAM-ID. INJOCA.
AUTHOR. KUNLAYAVEC.
DATE-COMPILED.
REMARKS. PROGRAM INQUIRY INFORMATION OF CREDIT GUA..
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
77  N-REMAIN                        PIC  9(05)  VALUE 0.
77  C-REMAIN                        PIC  9(05)  VALUE 0.
77  NO-GUA                          PIC  9(01)  VALUE 0.
77  MESS-L                          PIC  S9(4)  COMP VALUE +0.
77  MESS-OUT                        PIC  X(75).
77  TORP                            PIC  X(12)  VALUE SPACE.
77  ABLF                            PIC  9(01)  VALUE 0.
77  SAVEID1                         PIC  9(06)  VALUE 0.
77  SAVEID2                         PIC  9(06)  VALUE 0.
77  SAVEID3                         PIC  9(06)  VALUE 0.
77  SAVEID4                         PIC  9(06)  VALUE 0.
77  SAVEID5                         PIC  9(06)  VALUE 0.
77  GUAL1                           PIC  9(05)  VALUE 0.
77  GUAL2                           PIC  9(05)  VALUE 0.
77  GUAL3                           PIC  9(05)  VALUE 0.
77  GUAL4                           PIC  9(05)  VALUE 0.
77  GUAL5                           PIC  9(05)  VALUE 0.
77  NUMA                            PIC  9(04)  VALUE 0.
77  IDLOAN                          PIC  9(06)  VALUE 0.
77  AMTL                            PIC  9(06)  VALUE 0.
77  GUAL                            PIC  9(06)  VALUE 0.
77  EOSE                            PIC  9(01)  VALUE 0.
01  DFHBMSCA       COPY DFHBMSCA.
01  DFHAID         COPY DFHAID.
01  INPUTMI        COPY INPUTM.
01  OPGUAI         COPY OPGUA.
01  AMANI          COPY AMAN.
01  MAPERR1I       COPY MAPERR1.
01  K-DATA.
    05  K-YY                        PIC  9(04).
    05  K-MM                        PIC  9(02).
    05  K-ND                        PIC  9(04).
    05  K-SE                        PIC  9(01).
01  G-DATE.
    05  G-YY                        PIC  9(04).
    05  G-MM                        PIC  9(02).
    05  G-DD                        PIC  9(02).
01  E-DATE.
    05  E-DD                        PIC  9(02).
    05  FILLER1                     PIC  X(01)  VALUE '/'.
    05  E-MM                        PIC  9(02).
    05  FILLER                      PIC  X(01)  VALUE '/'.
    05  E-YY                        PIC  9(04).
01  KREC.
    05  KFEPP                       PIC  X(126).
01  ID-KEY5.
    05  ID5                         PIC  9(06).
    05  SEQ                         PIC  9(01).
    05  ACT5                        PIC  X(01).
01  ID-KEY1.
    05  ID1                         PIC  9(06).
    05  ACT1                        PIC  X(01).
01  MEMB-REC.
    05  ID-KEY-M.
        07  ID-MEMBER-M             PIC  9(06).
        07  A-I-M                   PIC  X(01).
    05  CHECK-M.
        07  P-I-M                   PIC  X(01).
        07  PEC-STATUS-M            PIC  X(01).
        07  F-L-M                   PIC  X(01).
    05  ID-MEMBER-OLD               PIC  9(04).
    05  THAI-NAME                   PIC  X(60).
    05  ENG-NAME                    PIC  X(40).
    05  SEX                         PIC  X(01).
    05  PAY-PLACE                   PIC  9(05).
    05  ID-PAY                      PIC  X(07).
```

```
    05  ID-OFFICE                    PIC  X(07).
    05  RE-MEMBER                    PIC  X(01).
    05  NAME-CLNO                    PIC  X(02).
    05  UPDATE-DATE-M                PIC  X(08).
    05  BLANK-M                      PIC  X(13).
01  STAT-REC.
    05  ID-KEY-S.
        07  ID-MEMBER-S              PIC  X(06).
        07  A-I-S                    PIC  X(02).
    05  CHECK-S.
        07  P-T-S                    PIC  X(01).
        07  REC-STATUS-S             PIC  X(01).
        07  F-L-S                    PIC  X(01).
    05  MEMBER-DATE                  PIC  9(08).
    05  MEMBER-AGE                   PIC  9(03).
    05  MAPITAL                      PIC  X(01).
    05  ID-SPOUSE                    PIC  9(06).
    05  DEPT-CODE                    PIC  X(03).
    05  OFFICE-CODE                  PIC  X(05).
    05  MEMBER-TYPE                  PIC  X(01).
    05  L-QUEUE                      PIC  9(02).
    05  AMT-LOAN                     PIC  9(05).
    05  L-STATUS.
        07  EMERGENCY                PIC  X(01).
        07  ORDINARY                 PIC  X(01).
        07  EXTRA                    PIC  X(01).
    05  AMT-GUA                      PIC  9(05).
    05  GUA-LIMIT-USED               PIC  9(05).
    05  GUA-MEM-STATUS               PIC  9(01).
    05  GUA-MEM-RECORD               PIC  9(01).
    05  SALARY                       PIC  9(05).
    05  C-INCOME                     PIC  9(04).
    05  XP-PRES-S                    PIC  9(04).
    05  O-PAYMENT                    PIC  9(05)V99.
    05  M-ADJUST                     PIC  9(03)V99.
    05  OUT-COND                     PIC  X(01).
    05  OUT-DATE                     PIC  X(08).
    05  UPDATE-DATE-S                PIC  X(06).
    05  BLANK-S                      PIC  X(29).
01  PERS-GUA-REC.
    05  GUA-MEM-KEY-P.
        07  ID-GUA-MEM-P             PIC  9(06).
        07  SEQUENCE-P2              PIC  9(01).
        07  A-I-P                    PIC  X(01).
    05  CHECK-P.
        07  P-T-P                    PIC  X(01).
        07  REC-STATUS-P             PIC  X(01).
        07  F-L-P                    PIC  X(01).
    05  ID-GUALO                     PIC  9(06).
    05  GUA-LIMIT                    PIC  9(05).
    05  ID-CO-GUA-KEY.
        07  ID-CO-GUA                PIC  9(06).
        07  SEQUENCE-P2              PIC  9(01).
    05  UPDATE-DATE-P                PIC  X(06).
    05  BLANK-P                      PIC  X(27).
01  DATA-REC.
    02  ID-TRAN-KEY.
      03  YEAR-TR                    PIC  9(04).
      03  MONTH-TR                   PIC  9(02).
      03  APPLY-NO                   PIC  9(04).
      03  SEQUENCE-NO                PIC  9(01).
    02  IDTR                         PIC  9(06).
    02  L-TYPE-TR                    PIC  X(01).
    02  PRINCIPAL-TR                 PIC  9(06).
    02  TOTAL-PERIOD-TR              PIC  9(03).
    02  ID-GUA-TR                    PIC  9(09).
    02  GUA-STAT-TR                  PIC  9(01).
    02  GUA-NUM-TR                   PIC  9(01).
    02  GUA-LIMIT-TR                 PIC  9(06).
    02  GUA-SHARE-TR                 PIC  9(06).
    02  GUA-DEPOSIT-TR               PIC  9(06).
    02  EVAL-DATE-TR                 PIC  9(08).
    02  LAND-AREA-TR                 PIC  9(04).
    02  EVAL-UNIT-TR                 PIC  9(06).
    02  EVAL-BUILDING-TR             PIC  9(07).
    02  FIRE-INSURANCE-TR            PIC  9(06).
    02  UPDATE-DATE-TR               PIC  9(08).
```

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

```
    02  BLANK-TI                    PIC  X(35).
PROCEDURE DIVISION.
P-START.
    EXEC CICS HANDLE CONDITION
              INVMPSZ  (P-INVMPSZ)
              MAPFAIL  (P-MAPFAIL)
              DSIDERR  (P-DSIDERR)
              LENGERR  (P-LENGERR)
              NOTFND   (P-NOTFND)
              NOSTG    (P-NOSTG)
              ERROR    (P-ERROR)  END-EXEC.
    EXEC CICS HANDLE AID
              CLEAR (P-CLEAR)
              ANYKEY  (P-ANYKEY)  END-EXEC.
P-READDATA.
    MOVE 900000000000 TO K-DATA.
    EXEC CICS READ INTO (DATA-REC)
                   DATASET ('CIMAFL')
                   RIDFLD (K-DATA)
                   KEYLENGTH (11) END-EXEC.
P-READDATA-EXIT. EXIT.
P-INPUT.
    MOVE LOW-VALUE TO INPUTMC.
    MOVE LOW-VALUE TO OPGUAO.
    MOVE LOW-VALUE TO MAPERRC.
    MOVE UPDATE-DATE-TP TO G-DATE.
    MOVE G-YY TO E-YY.
    MOVE G-MM TO E-MM.
    MOVE G-DD TO E-DD.
    MOVE E-DATE TO MDATEC.
    EXEC CICS SEND MAP ('INPUTM') FROM (INPUTMO)
                                  ERASE END-EXEC.
    EXEC CICS RECEIVE MAP ('INPUTM') INTO (INPUTMI) END-EXEC.
    MOVE KEYIDGI TO ID1.
    MOVE 0 TO ACT1.
P-READF1.
    EXEC CICS READ INTO (MEMB-REC)
                   DATASET ('MEMBFL')
                   RIDFLD (ID-KEY1)
                   KEYLENGTH (7)  END-EXEC.
    MOVE ENG-NAME TO NAMEGO.
    EXEC CICS SEND MAP ('INPUTM') FROM (INPUTMO) END-EXEC.
P-READF2.
    EXEC CICS READ INTO (STAT-REC)
                   DATASET ('STATFL')
                   RIDFLD (ID-KEY1)
                   KEYLENGTH (7)  END-EXEC.
P-READF2-EXIT. EXIT.
P-CS.
    IF CHECK-S = '000' OR '110' NEXT SENTENCE
    ELSE GO TO P-ERR.
P-MOVE.
    MOVE SPACE TO ERR1O.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
    MOVE 0 TO NC-GUA.
    MOVE MARITAL TO MARRYO.
    PERFORM P-SPOUSE THRU P-SPOUSE-EXIT.
    MOVE ID-SPOUSE TO IDSPOUO.
    MOVE AMT-GUA TO AMTGO.
    MOVE GUA-LIMIT-USED TO USEDO.
    COMPUTE N-REMAIN = AMT-GUA - GUA-LIMIT-USED.
    MOVE N-REMAIN TO REMNO.
    IF GUA-MEN-STATUS = 0
        MOVE ZERO TO IDGUAE1O GUALIM1O GUAREM1O
                     IDGUAE2O GUALIM2O GUAREM2O
                     IDGUAE3O GUALIM3O GUAREM3O
                     IDGUAE4O GUALIM4O GUAREM4O
                     IDGUAE5O GUALIM5O GUAREM5O
        MOVE SPACE TO TERP1O TERP2O TERP3O TERP4O TERP5O
        MOVE 'THIS ID-MEMBER DOES NOT GUARANTEE OTHER MEMBER' TO
        ERR1O
        PERFORM P-SERR THRU P-SERR-EXIT
        GO TO P-SEND1.
    MOVE 101 TO ID5.
    MOVE 1 TO SEG.
    MOVE '0' TO ACT5.
```

```
      MOVE ID-KEY5 TO GUA-REC-KEY-P.
      EXEC CICS STARTBR
                DATASET ('PERSFIL')
                RIDFLD (ID-KEY5)
                KEYLENGTH (8)
                END-EXEC.
****READ NEXT RECORD****
 P-READNEXT.
      EXEC CICS READNEXT
                INTO (PERS-GUA-REC)
                DATASET ('PERSFIL')
                RIDFLD (ID-KEY5)
                KEYLENGTH (8)
                END-EXEC.
      IF A-I-P = '0'
          IF CHECK-P = '000' OR '110' GO TO P-STATREC.
      IF A-I-P = '1' AND CHECK-P = '100' GO TO P-STATREC.
      GO TO P-READNEXT.
 P-STATREC.
      IF ID-GUA-MEM-P NOT = ID1 GO TO P-SPACE.
      IF PREVSI = 'Y'
          IF CHECK-P = '000' OR '100' MOVE '(PREVIOUS)' TO TORP
          GO TO P-READNEXT-EXIT
          ELSE GO TO P-READNEXT.
      IF CHECK-P = '000' MOVE '(PERMANENT)' TO TORP
          GO TO P-READNEXT-EXIT.
      IF CHECK-P = '110' MOVE '(TEMPORARY)' TO TORP
          GO TO P-READNEXT-EXIT.
      GO TO P-READNEXT.
 P-READNEXT-EXIT.
      ADD 1 TO NO-GUA.
      IF NO-GUA = 1   GO TO P-MOVE1.
      IF NO-GUA = 2   GO TO P-MOVE2.
      IF NO-GUA = 3   GO TO P-MOVE3.
      IF NO-GUA = 4   GO TO P-MOVE4.
      IF NO-GUA = 5   GO TO P-MOVE5.
      MOVE SPACE TO ERRO.
      MOVE '**GUARANTEE EVEN THAN 5 PERSONS**' TO ERRO.
      GO TO P-ERROR.
 P-MOVE1.
****MOVE INFORMATION ID-GUA1****
      MOVE ID-GUAED TO IDGUAE10 SAVEID1.
      MOVE GUA-LIMIT TO GUALIM10 GUAL1.
      COMPUTE C-REMAIN = AMT-GUA - GUA-LIMIT.
      MOVE C-REMAIN TO GUAREM1O.
      MOVE TORP TO TORP1O.
      GO TO P-READNEXT.
 P-MOVE2.
****MOVE INFORMATION ID-GUA2****
      MOVE ID-GUAED TO IDGUAE2O SAVEID2.
      MOVE GUA-LIMIT TO GUALIM2O GUAL2.
      COMPUTE C-REMAIN = C-REMAIN - GUA-LIMIT.
      MOVE C-REMAIN TO GUAREM2O.
      MOVE TORP TO TORP2O.
      GO TO P-READNEXT.
 P-MOVE3.
****MOVE INFORMATION ID-GUA3****
      MOVE ID-GUAED TO IDGUAE3O SAVEID3.
      MOVE GUA-LIMIT TO GUALIM3O GUAL3.
      COMPUTE C-REMAIN = AMT-GUA - GUA-LIMIT.
      MOVE C-REMAIN TO GUAREM3O.
      MOVE TORP TO TORP3O.
      GO TO P-READNEXT.
 P-MOVE4.
****MOVE INFORMATION ID-GUA4****
      MOVE ID-GUAED TO IDGUAE4O SAVEID4.
      MOVE GUA-LIMIT TO GUALIM4O GUAL4.
      COMPUTE C-REMAIN = AMT-GUA - GUA-LIMIT.
      MOVE C-REMAIN TO GUAREM4O.
      MOVE TORP TO TORP4O.
      GO TO P-READNEXT.
 P-MOVE5.
****MOVE INFORMATION ID-GUAED5****
      MOVE ID-GUAED TO IDGUAE5O SAVEID5.
      MOVE GUA-LIMIT TO GUALIM5O GUAL5.
      COMPUTE C-REMAIN = AMT-GUA - GUA-LIMIT.
      MOVE C-REMAIN TO GUAREM5O.
```

```
     MOVE TERP TO TERP5C.
 P-SPACE.
     EXEC CICS LOCK DATASET ('PERSFL') END-EXEC.
     IF NO-GUA = 1 GO TO P-MOVE04.
     IF NO-GUA = 2 GO TO P-MOVE03.
     IF NO-GUA = 3 GO TO P-MOVE02.
     IF NO-GUA = 4 GO TO P-MOVE01.
     GO TO P-SEND.
 P-READF2.
     IF CHECK-S = '110' NEXT SENTENCE
     ELSE GO TO ERR2.
     MOVE '1' TO ACT1.
     PERFORM P-READF2 THRU P-READF2-EXIT.
     IF CHECK-S = '100' GO TO P-MOVEC
     ELSE GO TO ERR21.
 P-MOVE04.
     MOVE ZERO TO IDGUAE2C.
     MOVE   0   TO GUALIM2C.
     MOVE   0   TO GUAREM2C.
     MOVE SPACE TO TERP2C.
 P-MOVE03.
     MOVE ZERO TO IDGUAL3C.
     MOVE   0   TO          GUALIM3C  GUAREM3C.
     MOVE SPACE TO TERP3C.
 P-MOVE02.
     MOVE ZERO TO IDGUAE4C.
     MOVE   0   TO          GUALIM4C  GUARE4EC.
     MOVE SPACE TO TERP4C.
 P-MOVE01.
     MOVE ZERO TO IDGUAE5C.
     MOVE   0   TO          GUALIM5C  GUARE45C.
     MOVE SPACE TO TERP5C.
*****SEND MAP OUTPUT*****
 P-SEND.
     MOVE SPACE TO MESS-OUT.
     MOVE SPACE TO PREVSC.
     MOVE ALL '=' TO LINE1C.
     EXEC CICS SEND MAP ('CPGUA')
                    FROM (CPGLAC)
                     END-EXEC.
     EXEC CICS RECEIVE MAP ('CPGUA') INTO (CPGUAI)
                     END-EXEC.
     IF PREVSI = 'Y' MOVE '0' TO ACT1
                     MOVE 1 TO ABEF
                     PERFORM P-READF2 THRU P-READF2-EXIT
                     IF CHECK-S = 'COJ' GO TO P-AMAN
                     ELSE GO TO P-READF21.
     IF PREVSI = 'N' AND ABEF = 1
         MOVE 0 TO ABEF
         MOVE 'C' TO ACT1
         GO TO P-READF2.
P-AMAN.
    MOVE LOW-VALUE TO AMANO.
    EXEC CICS SEND MAP ('AMAN') FROM (AMANO) END-EXEC.
    EXEC CICS RECEIVE MAP ('AMAN') INTO (AMANI) END-EXEC.
    IF YESNI = 'Y' NEXT SENTENCE
    ELSE GO TO P-STOP.
    MOVE NCAPI TO NUMA.
    IF KEYID1 = ID-SPOUSE PERFORM P-NO THRU P-SERR-EXIT
                          GO TO P-AMAN.
    MOVE KEYID1 TO IDLOAN.
    IF AMTWI > AMT-GUA PERFORM P-NO THRU P-SERR-EXIT
                       GO TO P-AMAN.
    MOVE AMTWI TO AMTL.
    IF IDLOAN = SAVEID1 OR SAVEID2 OR SAVEID3 OR SAVEID4
       OR SAVEID5  MOVE 0 TO EUSE
    ELSE MOVE 1 TO EUSE.
    IF AMTL > N-REMAIN NEXT SENTENCE
    ELSE GO TO P-YES.
    IF IDLOAN = SAVEID1 OR SAVEID2 OR SAVEID3 OR SAVEID4
       OR SAVEID5 NEXT SENTENCE
    ELSE PERFORM P-NO THRU P-SERR-EXIT
         GO TO P-AMAN.
```

```
P-CHECK.
    IF IDLCAN = SAVEID1 MOVE GUAL1 TO GUAL
        GO TO P-COMP.
    IF IDLCAN = SAVEID2 MOVE GUAL2 TO GUAL
        GO TO P-COMP.
    IF IDLCAN = SAVEID3 MOVE GUAL3 TO GUAL
        GO TO P-COMP.
    IF IDLCAN = SAVEID4 MOVE GUAL4 TO GUAL
        GO TO P-COMP.
    MOVE GUAL5 TO GUAL.
P-COMP.
    COMPUTE GUAL = GUAL + K-REMAIN.
    IF AMTL > GUAL PERFORM P-NO THRU P-SERR-EXIT
                   GO TO P-AMAN.
    GO TO P-YES.
P-SPOUSE.
    IF MARITAL = '1' MOVE ' (SINGLE)  ' TO MARITO
                     GO TO P-SPOUSE-EXIT.
    IF MARITAL = '2' MOVE ' (MARRIED) ' TO MARITO
                     GO TO P-SPOUSE-EXIT.
    IF MARITAL = '3' MOVE '(SEPARATED)' TO MARITO
                     GO TO P-SPOUSE-EXIT.
    IF MARITAL = '4' MOVE '   (WIDOW) ' TO MARITO
                     GO TO P-SPOUSE-EXIT.
    MOVE ' (DIVORCE) ' TO MARITO.
P-SPOUSE-EXIT. EXIT.
P-NO.
    MOVE SPACE TO LERRO.
    MOVE 'NOT ALLOW TO GUARANTY' TO LERRO.
P-SERR.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
P-SERR-EXIT. EXIT.
P-YES.
    MOVE SPACE TO LERRO.
    MOVE 'ALLOW TO GUARANTY' TO LERRO.
    PERFORM P-SERR THRU P-SERR-EXIT.
P-DATA.
    MOVE G-YY TO K-YY.
    MOVE G-MM TO K-MM.
    MOVE NUMA TO K-NO.
    MOVE 1 TO K-SE.
    MOVE ZERO TO KREC.
P-F9.
    EXEC CICS HANDLE CONDITION
                     NOTFND (P-WRITE) END-EXEC.
    EXEC CICS READ INTO (DATA-REC)
                   DATASET ('CIMAFL')
                   RIDFLD (K-DATA)
                   UPDATE
                   KEYLENGTH (11) END-EXEC.
    MOVE DATA-REC TO KREC.
    IF ID-GUA-TR = ZERO NEXT SENTENCE
    ELSE ADD 1 TO K-SE
        MOVE K-SE TO GUA-NUM-TR
        PERFORM P-REWRITE THRU P-REWRITE-EXIT
        GO TO P-F9.
P-FIRST.
    IF IDLCAN = 101 MOVE ZERO TO ID-GUA-TR
                    MOVE 0 TO GUA-STAT-TR
                    MOVE 0 TO GUA-NUM-TR
    ELSE MOVE 101 TO ID-GUA-TR
         MOVE EUSE TO GUA-STAT-TR
    MOVE K-SE TO GUA-NUM-TR.
    MOVE AMTL TO GUA-LIMIT-TR.
    MOVE USHARE1 TO GUA-SHARE-TR.
    MOVE USDEPO1 TO GUA-DEPOSIT-TR.
    MOVE G-DATE TO UPDATE-DATE-TR.
    MOVE ZERO TO EVAL-DATE-TR LAND-AREA-TR EVAL-UNIT-TR
                 FIRE-INSURANCE-TR
                 EVAL-BUILDING-TR.
P-FIRST-EXIT. EXIT.
P-REWRITE.
    EXEC CICS REWRITE DATASET ('CIMAFL')
                      FROM (DATA-REC) END-EXEC.
P-REWRITE-EXIT. EXIT.
P-READY.
    MOVE SPACE TO LERRO.
    MOVE 'PREPARED CARD IMAGE ALREADY' TO LERRO.
    PERFORM P-SERR THRU P-SERR-EXIT.
    EXEC CICS RETURN TRANSID ('INQ0') END-EXEC.
```

```
P-WRITE.
    MOVE KREC  TO DATA-REC.
    MOVE K-L  TO ID-TEMP-KEY.
    PERFORM T   TEST THRU P-FIRST-EXIT.
    EXEC CICS WRITE FILE (DATA-REC)
              DATASET ('CIMAFL')
              RIDFLD (K-DATA)
              KEYLENGTH (11) END-EXEC.
    GO TO P-READY.
P-ERR.
    MOVE 'ERROR.RECORD ID USEFUL NOT FOUND' TO LERRG.
    GO TO P-ERROR.
P-ENDF.
    MOVE 'END OF FILE' TO LERRG.
P-ERROR.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1) END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG') END-EXEC.
ERR2.
    MOVE 'RECORD ACTIVE TEMPORARY OF STAT-FILE NOT FOUND'
        TO LERRG.
    GO TO P-ERROR.
ERR21.
    MOVE 'RECORD INACTIVE TEMPORARY OF STAT-FILE NOT FOUND'
        TO LERRG.
    GO TO P-ERROR.
P-CLEAR.
    EXEC CICS RETURN  END-EXEC.
P-INVMPSZ.
    MOVE 16 TO MESS-L.
    MOVE 'INVALID MAP SIZE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-MAPFAIL.
    MOVE 8 TO MESS-L.
    MOVE 'MAP FAIL' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT  END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG')  END-EXEC.
P-DSIDERR.
    MOVE 17 TO MESS-L.
    MOVE 'DATA SET ID ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
          WAIT ERASE END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG')  END-EXEC.
P-LENGERR.
    MOVE 18 TO MESS-L.
    MOVE 'WRONG LENGTH ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG')  END-EXEC.
P-NOTFND.
    MOVE 16 TO MESS-L.
    MOVE 'RECORD NOT FOUND' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                       ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG')  END-EXEC.
P-NOSTG.
    MOVE 10 TO MESS-L.
    MOVE 'NO STORAGE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-ANYKEY.
    MOVE 32 TO MESS-L.
    MOVE 'DO NOT PRESS ENTER KEY TRY AGAIN' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE
                        WAIT END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG')  END-EXEC.
P-STOP.
    MOVE 'THIS SCREEN IS ONLY INQUIRY' TO LERRG.
    EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1) END-EXEC.
    EXEC CICS RETURN TRANSID ('INQG') END-EXEC.
    STOP RUN.
```

ค. ๖ โปรแกรมแสดงรายละเอียดของหลักทรัพย์ค้ำประกัน (โปรแกรม P05)

```
ID DIVISION.
PROGRAM-ID. INQLAND.
AUTHOR. KUN.
DATE-COMPILED.
REMARKS. PROGRAM INQUIRY INFORMATION OF GUARANTEE LAND.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
77  L-AMT                          PIC  9(06) VALUE 0.
77  ATOP                           PIC  9(6)  VALUE 0.
77  SAVE-ID                        PIC  9(06) VALUE 0.
77  NUM-MEN                        PIC  9(01) VALUE 0.
77  SAVE-PR                        PIC  9(06) VALUE 0.
77  C-AMT                          PIC  9(06) VALUE 0.
77  MESS-L                         PIC  S9(4) COMP VALUE +0.
77  MESS-OUT                       PIC  X(45).
77  RSTAT                          PIC  9(01)  VALUE 0.
77  GUA-STAT                       PIC  9(01)  VALUE 0.
77  SAVEID1                        PIC  9(06)  VALUE 0.
77  SAVEID2                        PIC  9(06)  VALUE 0.
77  SAVEID3                        PIC  9(06)  VALUE 0.
77  SAVEID4                        PIC  9(06)  VALUE 0.
77  SAVEID5                        PIC  9(06)  VALUE 0.
77  SAVEPP1                        PIC  9(06)  VALUE 0.
77  SAVEPR2                        PIC  9(06)  VALUE 0.
77  SAVEPR3                        PIC  9(06)  VALUE 0.
77  SAVEPR4                        PIC  9(06)  VALUE 0.
77  SAVEPR5                        PIC  9(06)  VALUE 0.
77  ID-MEM                         PIC  9(06)  VALUE 0.
77  L-TYPE                         PIC  X(01)  VALUE SPACE.
77  L-OPTN                         PIC  X(01)  VALUE SPACE.
77  TORP                           PIC  X(11)  VALUE SPACE.
77  APPLYNO                        PIC  9(04)  VALUE 0.
77  L-SECU                         PIC  9(07)  VALUE 0.
77  L-PRICE                        PIC  9(07)  VALUE 0.
77  L-SHARE                        PIC  9(06)  VALUE 0.
77  L-DEPO                         PIC  9(06)  VALUE 0.
01  ASSET-GUA-REC.
    05  GUA-LAND-KEY-A.
        07  ID-GUA-LAND-A          PIC  9(09).
        07  SEQUENCE-A1            PIC  9(01).
        07  A-I-A1                 PIC  X(01).
    05  CHECK-A.
        07  P-T-A                  PIC  X(01).
        07  REC-STATUS-A           PIC  X(01).
        07  F-L-A                  PIC  X(01).
    05  L-TYPE-A                   PIC  X(01).
    05  ID-MEMBER-A                PIC  9(06).
    05  PRINCIPAL-A                PIC  9(06).
    05  GUA-LIMIT-A                PIC  9(06).
    05  GUA-LAND-CO-KEY.
        07  ID-GUA-LAND-CO         PIC  9(09).
        07  SEQUENCE-A2            PIC  9(01).
    05  UPDATE-DATE-A              PIC  X(08).
    02  BLANK-A                    PIC  X(13).
01  LAND-STAT-REC.
    05  GUA-LAND-KEY-D.
        07  ID-GUA-LAND-D          PIC  9(09).
        07  A-I-D                  PIC  X(01).
    05  CHECK-D.
        07  P-T-D                  PIC  X(01).
        07  REC-STATUS-D           PIC  X(01).
        07  F-L-D                  PIC  X(01).
    05  L-TYPE-D                   PIC  X(01).
    05  GUA-LAND-RECORD            PIC  9(01).
    05  MEN-PER-LAND               PIC  9(01).
    05  EVAL-DATE                  PIC  9(08).
    05  LAND-AREA                  PIC  9(04).
    05  EVAL-UNIT                  PIC  9(06).
    05  EVAL-PRICE                 PIC  9(07).
    05  EVAL-BUILDING              PIC  9(07).
    05  FIRE-INSURANCE             PIC  9(06).
    05  AMT-GUA-SECUR              PIC  9(06).
```

```
           05  REMAIN-SEQ#                 PIC  9(06).
           05  UPDATE-DATE-D               PIC  X(06).
           05  BLANK-D                     PIC  X(54).
       01  DATA-REC.
           05  ID-TRAN-KEY.
             07  YEAR-TR                   PIC  9(04).
             07  MONTH-TR                  PIC  9(02).
             07  APPLY-TR                  PIC  9(04).
             07  SEQUENCE-TR               PIC  9(01).
           05  IDTR                        PIC  9(06).
           05  L-TYPE-TR                   PIC  X(01).
           05  PRINCIPAL-TR                PIC  9(06).
           05  TOTAL-PERIOD-TR             PIC  9(03).
           05  ID-GUA-TR                   PIC  9(09).
           05  GUA-STAT-TR                 PIC  9(01).
           05  GUA-NUM-TR                  PIC  9(02).
           05  GUA-LIMIT-TR                PIC  9(06).
           05  GUA-SHARE-TR                PIC  9(06).
           05  GUA-DEPOSIT-TR              PIC  9(06).
           05  EVAL-DATE-TR                PIC  9(08).
           05  LAND-AREA-TR                PIC  9(04).
           05  EVAL-UNIT-TR                PIC  9(06).
           05  EVAL-BUILDING-TR            PIC  9(07).
           05  FIRE-INSURANCE-TR           PIC  9(06).
           05  UPDATE-DATE-TR              PIC  9(08).
           05  BLANK-TR                    PIC  X(35).
       01  LAND-KEY.
           05  L-ID                        PIC  9(09).
           05  A-I                         PIC  X(01).
       01  ASSET-KEY.
           05  A-ID                        PIC  9(09).
           05  A-SEQ                       PIC  9(01).
           05  A-ACT.                      PIC  X(01).
       01  DFHBMSCA      COPY DFHBMSCA.
       01  DFHAID        COPY DFHAID.
       01  INPUTLI       COPY INPUTL.
       01  NEWEVAI       COPY NEWEVA.
       01  KDATAI        COPY KDATA.
       01  CLANDESI      COPY CLANDES.
       01  LGUAI         COPY LGUA.
       01  MAPERRII      COPY MAPERR1.
       01  EDIT-DATE.
           05  DD                          PIC  9(02).
           05  FILLER                      PIC  X(01)  VALUE '/'.
           05  MM                          PIC  9(02).
           05  FILLER                      PIC  X(01)  VALUE '/'.
           05  YY                          PIC  9(04).
       01  EV-DATE.
           05  EV-Y                        PIC  9(04).
           05  EV-M                        PIC  9(02).
           05  EV-D                        PIC  9(02).
       01  S-DATE.
           05  S-YY                        PIC  9(04).
           05  S-MM                        PIC  9(02).
           05  S-DD                        PIC  9(02).
       01  KEEP-REC.
           05  DATA1                       PIC  X(128).
       01  DATA-KEY.
           05  K-YY                        PIC  9(04).
           05  K-MM                        PIC  9(02).
           05  K-NO                        PIC  9(04).
           05  K-SE                        PIC  9(01).
       PROCEDURE DIVISION.
       P-START.
           EXEC CICS HANDLE CONDITION
                     INVMPSZ (P-INVMPSZ)
                     MAPFAIL (P-MAPFAIL)
                     DSIDERR (P-DSIDERR)
                     LENGERR (P-LENGERR)
                     NOTFND  (P-NOTFND)
                     NOSTG   (P-NOSTG)  END-EXEC.
           EXEC CICS HANDLE AID
                     CLEAR (P-CLEAR)
                     ANYKEY (P-ANYKEY)  END-EXEC.
       P-MOVEDATE.
           MOVE LOW-VALUE TO INPUTLO.
           MOVE 900000000000 TO DATA-KEY.
           EXEC CICS READ DATE (DATA-REC)
```

```
                DATASET ('CIMAFL')
                     RIDFLD (DATA-KEY)
                     KEYLENGTH (11) END-EXEC.
      MOVE UPDATE-DATE-TR TO S-DATE.
      MOVE S-DD TO DD.
      MOVE S-MM TO MM.
      MOVE S-YY TO YY.
      MOVE EDIT-DATE TO RDATEO.
  P-INPUT.
      EXEC CICS SEND MAP ('INPUTL') FROM (INPUTLO)
                                    ERASE END-EXEC.
      EXEC CICS RECEIVE MAP ('INPUTL') INTO (INPUTLI)  END-EXEC.
      MOVE LANDIDI TO L-ID.
      MOVE '0' TO A-I.
      EXEC CICS SEND MAP ('NEWEVA') MAPONLY FREEKB END-EXEC.
      EXEC CICS RECEIVE MAP ('NEWEVA') INTO (NEWEVAI) END-EXEC.
      IF KEYAI = 'Y' GO TO P-NEWLAND.
      MOVE LOW-VALUE TO CLANDESC.
      MOVE LOW-VALUE TO MAPERRIC.
  P-READF7.
      EXEC CICS READ INTO (LAND-STAT-REC)
                     DATASET ('LANDFL')
                     RIDFLD (LAND-KEY)
                     KEYLENGTH (10)  END-EXEC.
      IF A-I-D = '0'
          IF CHECK-D = '000' OR '110' GO TO P-MOVE.
      IF A-I-D = '1' AND CHECK-D = '100' NEXT SENTENCE
      ELSE GO TO P-ERR.
  P-MOVE.
      MOVE 0 TO NUM-MEN.
      MOVE L-TYPE-D TO L-TYPE LNTYPEO.
      IF L-TYPE-D = 'S' MOVE '= EXTRA (MORTGAGE)' TO SEXTRAO
      ELSE MOVE '= EXTRA (LAND)' TO SEXTRAO.
      MOVE MEN-PER-LAND TO MENNUMO.
      MOVE LAND-AREA TO LANDARO.
      MOVE EVAL-UNIT TO EVAPUTO.
      MOVE EVAL-PRICE TO EVAPRIO.
      MOVE EVAL-BUILDING TO EVABUDO.
      MOVE FIRE-INSURANCE TO FIREINO.
      MOVE EVAL-DATE TO EV-DATE.
      MOVE EV-D TO DD.
      MOVE EV-M TO MM.
      MOVE EV-Y TO YY.
      MOVE EDIT-DATE TO EVADATO.
      MOVE AMT-GUA-SECUR TO AMTGUAO.
      MOVE REMAIN-SECUR TO REMSECO.
      IF MEN-PER-LAND = 0 MOVE ZERO TO GUAEQIO PRINCIO SECUIO
                          OTHER1O
                          MOVE SPACE TO TORP1O
                          GO TO P-ZERO4.
****READ SEQUENTIAL OF FL****
    EXEC CICS HANDLE CONDITION
                ENDFILE (P-ENDF)  END-EXEC.
    MOVE L-ID TO A-ID.
    MOVE 1 TO A-SEQ.
    MOVE 'C' TO A-ACT.
    MOVE ASSET-KEY TO GUA-LAND-KEY-A.
    EXEC CICS STARTBR
                DATASET ('ASSTFL')
                RIDFLD (ASSET-KEY)
                END-EXEC.
P-READNEXT.
    EXEC CICS READNEXT
                INTO (ASSET-GUA-REC)
                DATASET ('ASSTFL')
                RIDFLD (ASSET-KEY)
                END-EXEC.
    IF A-I-A1 = '0'
        IF CHECK-A = '000' OR '110' GO TO P-MOVESTAT.
    IF A-I-A1 = '1' AND CHECK-A = '100' NEXT SENTENCE
    ELSE GO TO P-READNEXT.
P-MOVESTAT.
    IF ID-GUA-LAND-A NOT = L-ID GO TO P-SPACE.
    IF CSTATI = 'Y'
        IF CHECK-A = '000' OR '100' MOVE SPACE TO TORP
                                    MOVE '(PREVIOUS)' TO TORP
                                    GO TO P-READNEXT-EXIT
                                    ELSE GO TO P-READNEXT.
```

```
     IF CHECK-A = '000' MOVE '(PERMANENT)' TO TORP
                                          GO TO P-READNEXT-EXIT.
     IF CHECK-A = '110' MOVE '(TEMPORARY)' TO TORP
                                          GO TO P-READNEXT-EXIT.
     MOVE SPACE TO TORP.
     MOVE 'STATUS RECORD OF ASST-FILE ERROR' TO TORRO.
     GO TO P-SEND.
 P-READNEXT-EXIT.
     ADD 1 TO NUM-MEN.
     IF NUM-MEN = 1  GO TO P-MOVE1.
     IF NUM-MEN = 2  GO TO P-MOVE2.
     IF NUM-MEN = 3  GO TO P-MOVE3.
     IF NUM-MEN = 4  GO TO P-MOVE4.
     IF NUM-MEN = 5  GO TO P-MOVE5.
     MOVE SPACE TO TERRO.
     MOVE '**GUARANTEE OVER THAN 5 PERSONS**' TO TERRO.
     GO TO P-SEND.
 P-MOVE1.
*****MOVE INFORMATION ID1*****
     MOVE ID-MEMBER-A TO SAVEID1 GUAED1C.
     MOVE TORP TO TORP1C.
     MOVE PRINCIPAL-A TO PRINC1C.
     MOVE GUA-LIMIT-A TO SECU1C SAVEPR1.
     COMPUTE ATOR = PRINCIPAL-A - GUA-LIMIT-A.
     MOVE ATOR TO OTHER1C.
     GO TO P-READNEXT.
 P-MOVE2.
*****MOVE INFORMATION ID2*****
     MOVE ID-MEMBER-A TO SAVEID2 GUAED2C.
     MOVE TORP TO TORP2 .
     MOVE PRINCIPAL-A TO PRINC2C.
     MOVE GUA-LIMIT-A TO SECU2C SAVEPR2.
     COMPUTE ATOR = PRINCIPAL-A - GUA-LIMIT-A.
     MOVE ATOR TO OTHER2C.
     GO TO P-READNEXT.
 P-MOVE3.
*****MOVE INFORMATION ID3*****
     MOVE ID-MEMBER-A TO SAVEID3 GUAED3C.
     MOVE TORP TO TORP3C.
     MOVE PRINCIPAL-A TO PRINC3C.
     MOVE GUA-LIMIT-A TO SECU3C SAVEPR3.
     COMPUTE ATOR = PRINCIPAL-A - GUA-LIMIT-A.
     MOVE ATOR TO OTHER3C.
     GO TO P-READNEXT.
 P-MOVE4.
*****MOVE INFORMATION ID4*****
     MOVE ID-MEMBER-A TO SAVEID4 GUAED4C.
     MOVE TORP TO TORP4C.
     MOVE PRINCIPAL-A TO PRINC4C.
     MOVE GUA-LIMIT-A TO SECU4C SAVEPR4.
     COMPUTE ATOR = PRINCIPAL-A - GUA-LIMIT-A.
     MOVE ATOR TO OTHER4C.
     GO TO P-READNEXT.
 P-MOVE5.
*****MOVE INFORMATION ID5*****
     MOVE ID-MEMBER-A TO SAVEID5 GUAED5C.
     MOVE TORP TO TORP5C.
     MOVE PRINCIPAL-A TO PRINC5C.
     MOVE GUA-LIMIT-A TO SECU5C SAVEPR5.
     COMPUTE ATOR = PRINCIPAL-A - GUA-LIMIT-A.
     MOVE ATOR TO OTHER5C.
 P-SPACE.
     EXEC CICS ENDBR DATASET ('ASSIFL') END-EXEC.
     IF NUM-MEN = 1 GO TO P-ZERO4.
     IF NUM-MEN = 2 GO TO P-ZERO3.
     IF NUM-MEN = 3 GO TO P-ZERO2.
     IF NUM-MEN = 4 GO TO P-ZERO1.
     GO TO P-SENDMAP.
 P-ZERO4.
     MOVE ZERO TO GUAED2C PRINC2C.
     MOVE 0 TO SECU2C OTHER2C.
     MOVE SPACE TO TORP2C.
 P-ZERO3.
     MOVE ZERO  TO GUAED3C PRINC3C.
     MOVE 0 TO SECU3C OTHER3C.
```

```
           MOVE SPACE TO TOKP2O.
       P-ZERO2.
           MOVE ZERO  TO GUAED4O PRINC4O.
           MOVE O TO SECU4O OTHER4O.
           MOVE SPACE TO TOKP4O.
       P-ZERO1.
           MOVE ZERO  TO GUAED5O PRINC5O.
           MOVE O TO SECU5O OTHER5O.
           MOVE SPACE TO TOKP5O.
           MOVE ALL '=' TO LINE1O.
       P-ERROR.
           MOVE SPACE TO LERRO.
           EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
       P-ERROR-EXIT.      EXIT.
      ******************************************************************
      ****KEY DATA FROM APPLICATION FORM****
      ******************************************************************

       P-SENDMAP.
           MOVE SPACE TO LBLANKO.
           MOVE SPACE TO CSTATO.
           EXEC CICS SEND MAP ('GLANDES')
                          FROM (GLANDESO)
                          END-EXEC.
           EXEC CICS RECEIVE MAP ('GLANDES') INTO (GLANDESI) END-EXEC.
           IF CSTATI = 'Y' MOVE 1 TO KSTAT
                           MOVE '1' TO A-I
                           GO TO P-READF7.
           IF CSTATI = 'N' AND KSTAT = 1 MOVE 0 TO KSTAT
                           MOVE '0' TO A-I
                           GO TO P-READF7.
       P-SENDL.
           PERFORM P-ERROR THRU P-ERROR-EXIT.
           MOVE LOW-VALUE TO LGUAO.
           EXEC CICS SEND MAP ('LGUA') FROM (LGUAO) END-EXEC.
           EXEC CICS RECEIVE MAP ('LGUA') INTO (LGUAI) END-EXEC.
           IF ANSWERI = 'Y' NEXT SENTENCE
           ELSE GO TO P-STOP.
       P-LOAN.
           MOVE NOAPLI TO APPLYNO.
           MOVE IDMEMI TO ID-MEM.
           MOVE AMTGSI TO L-AMT.
           MOVE LSHAREI TO L-SHARE.
           MOVE LDEPOSI TO L-DEPO.
       P-SENDL-EXIT. EXIT.
       P-CHECKREM.
           IF ID-MEM = SAVEID1 OR SAVEID2 OR SAVEID3 OR SAVEID4
              OR SAVEID5  MOVE 0 TO GUA-STAT
           ELSE MOVE 1 TO GUA-STAT.
           IF L-AMT > REMAIN-SECUR GO TO P-CHECKID.
       P-ALLOW.
           MOVE SPACE TO LERRO.
           MOVE 'ALLOW TO USE THIS LAND' TO LERRO
           EXEC CICS SEND MAP ('MAPERR1') FROM (MAPERR1O) END-EXEC.
           GO TO P-USE.
       P-CHECKID.
           IF ID-MEM = SAVEID1
               MOVE SAVEID1 TO SAVE-ID
               MOVE SAVEPR1 TO SAVE-PR
               GO TO P-COMPUTE.
           IF ID-MEM = SAVEID2
               MOVE SAVEID2 TO SAVE-ID
               MOVE SAVEPR2 TO SAVE-PR
               GO TO P-COMPUTE.
           IF ID-MEM = SAVEID3
               MOVE SAVEID3 TO SAVE-ID
               MOVE SAVEPR3 TO SAVE-PR
               GO TO P-COMPUTE.
           IF ID-MEM = SAVEID4
               MOVE SAVEID4 TO SAVE-ID
               MOVE SAVEPR4 TO SAVE-PR
           GO TO P-COMPUTE.
           IF ID-MEM = SAVEID5
               MOVE SAVEID5 TO SAVE-ID
               MOVE SAVEPR5 TO SAVE-PR
               GO TO P-COMPUTE.
```

```
P-NOTALLOW.
    MOVE 'NOT ALLOW TO USE THIS LAND' TO LERR-1.
    PERFORM P-SERR THRU P-SERR-EXIT.
    GO TO P-SEND1.
P-COMPUTE.
    COMPUTE C-AMT = SAVE-PR + REMAIN-SECUR.
    IF L-AMT > C-AMT  GO TO P-NOTALLOW
    ELSE
         GO TO P-ALLOW.
P-USE.
    EXEC CICS HANDLE CONDITION
              ENDFILE (P-ENDF)
              DUPREC (P-DUPREC)
              NOTFND (P-WRITE)  END-EXEC.
    MOVE 0 TO K-SE.
    MOVE ZERO TO KEEP-REC.
    MOVE ZERO TO DATA-REC.
P-MOVEKEY.
    MOVE S-YY TO K-YY.
    MOVE S-MM TO K-MM.
    MOVE APPLYNO TO K-NO.
    ADD 1 TO K-SE.
    MOVE DATA-KEY TO ID-TRAN-KEY.
P-READDATA.
    EXEC CICS READ INTO (DATA-REC)
              DATASET ('CIMAFL')
                   RIDFLD (DATA-KEY)
                   UPDATE
                   KEYLENGTH (11)  END-EXEC.
P-HAVE.
    IF IDFR = ID-MEM NEXT SENTENCE
    ELSE  MOVE 'RECORD OF DATA-FILE ERROR' TO LERR-1
          GO TO P-SERR.
    IF ID-GUA-TR = ZERO NEXT SENTENCE
    ELSE ADD 1 TO GUA-NUM-TR
         PERFORM P-REWRITE THRU P-REWRITE-EXIT
         MOVE DATA-REC TO KEEP-REC
         GO TO P-MOVEKEY.
    MOVE L-ID TO ID-GUA-TR.
    MOVE L-TYPE TO L-TYPE-TR.
    MOVE GUA-STAT TO GUA-STAT-TR.
    MOVE 1 TO GUA-NUM-TR.
    MOVE L-AMT TO GUA-LIMIT-TR.
    MOVE L-SHARE TO GUA-SHARE-TR.
    MOVE L-DEPC TO GUA-DEPOSIT-TR.
    MOVE S-DATE TO UPDATE-DATE-TR.
P-LDETAIL.
    MOVE EVAL-DATE TO EVAL-DATE-TR.
    MOVE LAND-AREA TO LAND-AREA-TR.
    MOVE EVAL-UNIT TO EVAL-UNIT-TR.
    MOVE EVAL-BUILDING TO EVAL-BUILDING-TR.
    MOVE SPACE TO BLANK-TR.
P-LDETAIL-EXIT. EXIT.
P-REWRITE.
    EXEC CICS REWRITE DATASET ('CIMAFL')
                      FROM (DATA-REC) END-EXEC.
P-REWRITE-EXIT. EXIT.
P-READY.
    PERFORM P-READY THRU P-READY-EXIT.
    EXEC CICS RETURN TRANSID ('PLAN')  END-EXEC.
P-WRITE.
    MOVE KEEP-REC TO DATA-REC.
    MOVE DATA-KEY TO ID-TRAN-KEY.
    MOVE GUA-STAT TO GUA-STAT-TR.
    MOVE L-ID TO ID-GUA-TR.
    MOVE L-AMT TO GUA-LIMIT-TR.
    MOVE L-SHARE TO GUA-SHARE-TR.
    MOVE L-DEPC TO GUA-DEPOSIT-TR.
    PERFORM P-LDETAIL THRU P-LDETAIL-EXIT.
    EXEC CICS WRITE FROM (DATA-REC)
              DATASET ('CIMAFL')
                      RIDFLD (DATA-KEY)
                      KEYLENGTH (11) END-EXEC.
    PERFORM P-READY THRU P-READY-EXIT.
    EXEC CICS RETURN TRANSID ('PLAN')  END-EXEC.
*****************************************************************
*****KEY NEW LAND*****
```

```
*****************************************************************

P-NEWLAND.
    MOVE LOW-VALUE TO KDATAO.
    MOVE ALL '=' TO LINE2O.
    EXEC CICS SEND MAP ('KDATA') FROM (KDATAO) END-EXEC.
    EXEC CICS RECEIVE MAP ('KDATA') INTO (KDATAI) END-EXEC.
    MOVE SPACE TO LAND-STAT-REC.
    MOVE KLTYPEI TO L-TYPE-D.
    MOVE KAREAI TO LAND-AREA.
    MOVE KUNITI TO EVAL-UNIT.
    COMPUTE L-PRICE = (LAND-AREA * EVAL-UNIT).
    MOVE L-PRICE TO KPRICEO.
    MOVE KBUILDI TO EVAL-BUILDING.
    MOVE KFIREI TO FIRE-INSURANCE.
    IF L-TYPE-D = '3'
        COMPUTE L-SECU = (EVAL-BUILDING + L-PRICE) * 0.5
    ELSE COMPUTE L-SECU = (EVAL-BUILDING + L-PRICE) * 0.8.
    MOVE L-SECU TO GSECURO.
    MOVE KEDATEI TO S-DD.
    MOVE KEMONI TO S-MM.
    MOVE KEYEARI TO S-YY.
    MOVE S-DATE TO EVAL-DATE.
    MOVE EVAL-DATE-TR TO S-DATE.
    MOVE 1 TO GOA-STAT.
    EXEC CICS SEND MAP ('KDATA') FROM (KDATAO) END-EXEC.
    PERFORM P-SENDL THRU P-SENDL-EXIT.
    IF L-AMT > L-SECU GO TO P-NOTALLOW.
    GO TO P-ALLOW.
P-DUPREC.
    MOVE 'WRITE ERROR BECAUSE DUPLICATE RECORD' TO LERRO.
    GO TO P-SERR.
P-ERR.
    MOVE SPACE TO LERRO.
    MOVE 'RECORD USEFUL NOT FOUND' TO LERRO.
    GO TO P-SERR.
P-SERR.
    EXEC CICS SEND MAP ('MAPLRR1') FROM (MAPLRR1O) END-EXEC.
P-SERR-EXIT.   EXIT.
P-CLEAR.
    EXEC CICS RETURN END-EXEC.
P-READY.
    MOVE SPACE TO LERRO.
    MOVE '***PREPARED CARD IMAGE ALREADY***' TO LERRO.
    PERFORM P-SERR THRU P-SERR-EXIT.
P-READY-EXIT. EXIT.
P-RETURN.
    EXEC CICS RETURN TRANSID ('PLAN') END-EXEC.
P-INVMPSZ.
    MOVE 16 TO MESS-L.
    MOVE 'INVALID MAP SIZE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-MAPFAIL.
    MOVE 8 TO MESS-L.
    MOVE 'MAP FAIL' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-DSIDERR.
    MOVE 17 TO MESS-L.
    MOVE 'DATA SET ID ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-LENGERR.
    MOVE 18 TO MESS-L.
    MOVE 'WRONG LENGTH ERROR' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN END-EXEC.
```

```
P-ANYKEY.
    MOVE 32 TO MESS-L.
    MOVE 'DO NOT PRESS ENTER ONLY TRY AGAIN' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN TRANSID ('PLAN')  END-EXEC.
P-ENDF.
    MOVE 'END OF FILE' TO LERRO.
    GO TO P-SERR.
P-NOTFND.
    MOVE SPACE TO LERRO.
    MOVE '**RECORD NOT FOUND**' TO LERRO.
    EXEC CICS SEND MAP ('MAPLRR1') FROM (MAPERR1O) END-EXEC.
    GO TO P-SEND1.
P-NOSTG.
    MOVE 10 TO MESS-L.
    MOVE 'NO STORAGE' TO MESS-OUT.
    EXEC CICS SEND TEXT FROM (MESS-OUT)
                        LENGTH (MESS-L)
                        ERASE WAIT  END-EXEC.
    EXEC CICS RETURN END-EXEC.
P-STOP.
    MOVE SPACE TO LERRO.
    MOVE 'THIS SCREEN IS ONLY INQUIRY'  TO LERRO.
    PERFORM P-SERR THRU P-SERR-EXIT.
    EXEC CICS RETURN TRANSID ('PLAN') END-EXEC.
    STOP RUN.
```

ค.๗ โปรแกรมอัพเดทการเตรียมการขอกู้ใหม่ (โปรแกรม P06)

```
ID  DIVISION.
PROGRAM-ID. PREPAREL.
DATE-COMPILED. 26/07/82
REMARKS. PROGRAM PREPARE TEMPORARY LOAN FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370-138.
OBJECT-COMPUTER. IBM-370-138.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT STAT-FILE ASSIGN TO SYS015-DA-3340-STATFL
        PASSWORD IS PSWD-S
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS ID-KEY-S
        FILE STATUS IS CHKS.
    SELECT LOAN-FILE ASSIGN TO SYS017-DA-3340-LOANFL
        PASSWORD IS PSWD-L
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS ID-KEY-L
        FILE STATUS IS CHKL.
    SELECT PERS-GUA-FILE ASSIGN TO SYS018-DA-3340-PERSFL .
        PASSWORD IS PSWD-P
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS GUA-MEN-KEY-P
        FILE STATUS IS CHKP.
    SELECT ASSET-GUA-FILE ASSIGN TO SYS019-DA-3340-ASSTFL
        PASSWORD IS PSWD-A
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS GUA-LAND-KEY-A
        FILE STATUS IS CHKA.
    SELECT LAND-STAT-FILE ASSIGN TO SYS020-DA-3340-LANDFL
        PASSWORD IS PSWD-D
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS GUA-LAND-KEY-D
        FILE STATUS IS CHKD.
    SELECT TEMP-LOAN-FILE ASSIGN TO SYS021-DA-3340-TEMPFL
        PASSWORD IS PSWD-T
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS ID-KEY-T
        FILE STATUS IS CHKT.
    SELECT DATA-FILE ASSIGN TO SYS022-DA-3340-CIMAFL
        PASSWORD IS PSWD-C
        ORGANIZATION IS INDEXED
        ACCESS IS DYNAMIC
        RECORD KEY IS ID-TRAN-KEY
        FILE STATUS IS CHKC.
    SELECT PRINT-FILE ASSIGN TO SYS006-UR-1403-S.
DATA DIVISION.
FILE SECTION.
FD  STAT-FILE
    LABEL RECORD IS STANDARD
    DATA RECORD IS STAT-REC.
01  STAT-REC.
    02  ID-KEY-S.
        03  ID-MEMBER-S          PIC  9(06).
        03  A-I-S                PIC  X(01).
    02  CHECK-S.
        03  P-T-S                PIC  X(01).
        03  REC-STATUS-S         PIC  X(01).
        03  F-L-S                PIC  X(01).
    02  MEMBER-DATE              PIC  9(08).
    02  MEMBER-AGE               PIC  9(03).
    02  MARITAL                  PIC  X(01).
    02  ID-SPOUSE                PIC  9(06).
    02  DEPT-CODE                PIC  X(03).
    02  OFFICE-CODE              PIC  X(03).
    02  MEMBER-TYPE              PIC  X(01).
    02  L-QUEUE                  PIC  9(02).
    02  AMT-LOAN                 PIC  9(05).
```

```
    02  L-STATUS.
        03  EMERGENCY              PIC  X(01).
        03  ORDINARY               PIC  X(01).
        03  EXTRA                  PIC  X(01).
    02  AMT-GUA                    PIC  9(05).
    02  GUA-LIMIT-USED             PIC  9(05).
    02  GUA-MEN-STATUS             PIC  9(01).
    02  GUA-MEN-RECORD             PIC  9(01).
    02  SALARY                     PIC  9(05).
    02  O-INCOME                   PIC  9(04).
    02  XF-PRES-S                  PIC  9(04).
    02  O-PAYMENT                  PIC  9(5)V99.
    02  M-ADJUST                   PIC  9(3)V99.
    02  OUT-COND                   PIC  X(01).
    02  OUT-DATE                   PIC  X(08).
    02  UPDATE-DATE-S              PIC  X(08).
    02  BLANK-S                    PIC  X(29).
FD  LOAN-FILE
    LABEL RECORD IS STANDARD
    DATA RECORD IS LOAN-REC.
01  LOAN-REC.
    02  ID-KEY-L.
        03  ID-MEMBER-L            PIC  9(06).
        03  L-TYPE-L               PIC  X(01).
        03  A-I-L1                 PIC  X(01).
    02  CHECK-L.
        03  P-T-L1                 PIC  X(01).
        03  REC-STATUS-L           PIC  X(01).
        03  F-L-L                  PIC  X(01).
    02  L-AGREE-NO                 PIC  9(07).
    02  L-DATE                     PIC  9(06).
    02  PRINCIPAL-L                PIC  9(06).
    02  L-FLAG                     PIC  X(01).
    02  PRINCE-F-L-PAID            PIC  9(05).
    02  PRINC-MONTHLY-PAID         PIC  9(05).
    02  DEBT-REMAIN                PIC  9(06).
    02  TOTAL-PERIOD               PIC  9(03).
    02  PERIOD                     PIC  9(03).
    02  INT-F-L-PAID               PIC  9(04)V99.
    02  INT-PRES-PAID              PIC  9(04)V99.
    02  INT-DECRE                  PIC  9(03)V99.
    02  INT-RATE                   PIC  9(02)V9.
    02  INTX                       PIC  9(05)V99.
    02  GUA-MEN-KEY-L.
        03  ID-GUA-MEN-L           PIC  9(06).
        03  SEQUENCE-L2            PIC  9(01).
    02  GUA-LAND-KEY-L.
        03  ID-GUA-LAND-L          PIC  9(09).
        03  SEQUENCE-L3            PIC  9(01).
    02  GUA-LAND-NUMB              PIC  9(01).
    02  GUA-SHARE                  PIC  9(06).
    02  GUA-DEPOSIT                PIC  9(06).
    02  UPDATE-DATE-L              PIC  X(08).
    02  BLANK-L                    PIC  X(08).
FD  PERS-GUA-FILE
    LABEL RECORD IS STANDARD
    DATA RECORD IS PERS-GUA-REC.
01  PERS-GUA-REC.
    02  GUA-MEN-KEY-P.
        03  ID-GUA-MEN-P           PIC  9(06).
        03  SEQUENCE-P1            PIC  9(01).
        03  A-I-P                  PIC  X(01).
    02  CHECK-P.
        03  P-T-P                  PIC  X(01).
        03  REC-STATUS-P           PIC  X(01).
        03  F-L-P                  PIC  X(01).
    02  ID-GUAED                   PIC  9(06).
    02  GUA-LIMIT                  PIC  9(05).
    02  ID-CO-GUA-KEY.
        03  ID-CO-GUA              PIC  9(06).
        03  SEQUENCE-P2            PIC  9(01).
    02  UPDATE-DATE-P              PIC  X(08).
    02  BLANK-P                    PIC  X(27).
FD  ASSET-GUA-FILE
    LABEL RECORD IS STANDARD
    DATA RECORD IS ASSET-GUA-REC.
01  ASSET-GUA-REC.
    02  GUA-LAND-KEY-A.
        03  ID-GUA-LAND-A          PIC  9(09).
        03  SEQUENCE-A1            PIC  9(01).
        03  A-I-A1                 PIC  X(01).
    02  CHECK-A.
        03  P-T-A                  PIC  X(01).
```

```
           03  REC-STATUS-A          PIC  X(01).
           03  F-L-A                 PIC  X(01).
       02  L-TYPE-A                  PIC  X(01).
       02  ID-MEMBER-A               PIC  9(06).
       02  PRINCIPAL-A               PIC  9(06).
       02  GUA-LIMIT-A               PIC  9(06).
       02  GUA-LAND-CO-KEY.
           03  ID-GUA-LAND-CO        PIC  9(09).
           03  SEQUENCE-A2           PIC  9(01).
       02  UPDATE-DATE-A             PIC  X(08).
       02  BLANK-A                   PIC  X(13).
FD     LAND-STAT-FILE
       LABEL RECORD IS STANDARD
       DATA RECORD IS LAND-STAT-REC.
       01  LAND-STAT-REC.
           02  GUA-LAND-KEY-D.
           03  ID-GUA-LAND-D         PIC  9(09).
           03  A-I-D                 PIC  X(01).
       02  CHECK-D.
           03  P-T-D                 PIC  X(01).
           03  REC-STATUS-D          PIC  X(01).
           03  F-L-D                 PIC  X(01).
       02  L-TYPE-D                  PIC  X(01).
       02  GUA-LAND-RECORD           PIC  9(01).
       02  MEN-PER-LAND              PIC  9(01).
       02  EVAL-DATE                 PIC  9(08).
       02  LAND-AREA                 PIC  9(04).
       02  EVAL-UNIT                 PIC  9(06).
       02  EVAL-PRICE                PIC  9(07).
       02  EVAL-BUILDING             PIC  9(07).
       02  FIRE-INSURANCE            PIC  9(06).
       02  AMT-GUA-SECUR             PIC  9(06).
       02  REMAIN-SECUR              PIC  9(06).
       02  UPDATE-DATE-D             PIC  X(08).
       02  BLANK-D                   PIC  X(54).
FD     TEMP-LOAN-FILE
       LABEL RECORD IS STANDARD
       DATA RECORD IS TEMP-LOAN-REC.
01     TEMP-LOAN-REC.
       02  ID-KEY-T.
           03  ID-MEMBER-T           PIC  9(06).
           03  L-TYPE-T              PIC  X(01).
           03  A-I-T1                PIC  X(01).
       02  CHECK-T.
           03  P-T-T1                PIC  X(01).
           03  REC-STATUS-T          PIC  X(01).
           03  F-L-T                 PIC  X(01).
       02  L-AGREE-NO-T              PIC  9(07).
       02  L-DATE-T                  PIC  9(08).
       02  PRINCIPAL-T               PIC  9(06).
       02  L-FLAG-T                  PIC  X(01).
       02  PRINC-F-L-PAID-T          PIC  9(05).
       02  PRINC-MONTHLY-PAID-T      PIC  9(05).
       02  DEBT-REMAIN-T             PIC  9(06).
       02  TOTAL-PERIOD-T            PIC  9(03).
       02  PERIOD-T                  PIC  9(03).
       02  INT-F-L-PAID-T            PIC  9(04)V99.
       02  INT-PRES-PAID-T           PIC  9(04)V99.
       02  INT-DECRE-T               PIC  9(03)V99.
       02  INT-RATE-T                PIC  9(02)V9.
       02  INTX-T                    PIC  9(05)V99.
       02  GUA-MEN-KEY-T.
           03  ID-GUA-MEN-T          PIC  9(06).
           03  SEQUENCE-T2           PIC  9(01).
       02  GUA-LAND-KEY-T.
           03  ID-GUA-LAND-T         PIC  9(09).
           03  SEQUENCE-T3           PIC  9(01).
       02  GUA-LAND-NUMB-T           PIC  9(01).
       02  GUA-SHARE-T               PIC  9(06).
       02  GUA-DEPOSIT-T             PIC  9(06).
       02  UPDATE-DATE-T             PIC  X(08).
       02  BLANK-T                   PIC  X(08).
FD     DATA-FILE
       LABEL RECORD IS STANDARD
       DATA RECORD IS DATA-REC.
01     DATA-REC.
       02  ID-TRAN-KEY               PIC  9(11).
       02  FILLER                    PIC  X(117).
```

```
FD  PRINT-FILE
    RECORDING MODE IS F
    LABEL RECORD IS OMITTED
    DATA RECORD IS PRINT-REC.
01  PRINT-REC                      PIC  X(133).
WORKING-STORAGE SECTION.
77  PSWD-S                         PIC  X(09)  VALUE 'PSWDSTAT'.
77  PSWD-L                         PIC  X(08)  VALUE 'PSWDLOAN'.
77  PSWD-P                         PIC  X(08)  VALUE 'PSWDPERS'.
77  PSWD-A                         PIC  X(08)  VALUE 'PSWDASST'.
77  PSWD-D                         PIC  X(08)  VALUE 'PSWDLAND'.
77  PSWD-T                         PIC  X(08)  VALUE 'PSWDTEMP'.
77  PSWD-C                         PIC  X(08)  VALUE 'PSWDCIMA'.
77  ID-TRAN                        PIC  9(06)  VALUE 0.
77  TCAL-AMTGUA                    PIC  9(07)  VALUE 0.
77  CAL-AMTGUA                     PIC  9(06)  VALUE 0.
77  CL-QUEUE                       PIC  9(02)  VALUE 0.
77  C-SEQUENCE                     PIC  9(01)  VALUE 0.
77  C-REMAIN                       PIC  S9(06) VALUE +0.
77  CAL-GUA-USED                   PIC  9(05)  VALUE 0.
77  CAL-MEN                        PIC  9(01)  VALUE 0.
77  CAL-RECORD                     PIC  9(01)  VALUE 0.
77  CX                             PIC  9(01)  VALUE 0.
77  COUNT-TRAN                     PIC  9(03)  VALUE 0.
77  PAGE-NO                        PIC  9(03)  VALUE 0.
77  LINE-C                         PIC  9(02)  VALUE 0.
77  DUM                            PIC  9(01)  VALUE 0.
77  DUM-ID                         PIC  9(06)  VALUE 0.
77  DUM-LAND                       PIC  9(09)  VALUE 0.
77  K-ORDER                        PIC  9(10)  VALUE 0.
77  SUM-LIMIT                      PIC  9(07)  VALUE 0.
77  SW-D                           PIC  9(01)  VALUE 0.
77  SW-PAGE                        PIC  9(01)  VALUE 0.
77  SW-1                           PIC  X(01).
77  SW-2                           PIC  X(01).
77  SW-A                           PIC  9(01).
77  CHKS                           PIC  X(02).
77  CHKL                           PIC  X(02).
77  CHKP                           PIC  X(02).
77  CHKA                           PIC  X(02).
77  CHKD                           PIC  X(02).
77  CHKT                           PIC  X(02).
77  CHKC                           PIC  X(02)  VALUE '0'.
01  STAT-REC-W1.
    02  ID-KEY-SW1.
        03  ID-MEMBER-SW1          PIC  9(06).
        03  A-I-SW1                PIC  X(01).
    02  CHECK-SW1.
        03  P-T-SW1                PIC  X(01).
        03  REC-STATUS-SW1         PIC  X(01).
        03  F-L-SW1                PIC  X(01).
    02  MEMBER-DATE-W1             PIC  9(08).
    02  MEMBER-AGE-W1              PIC  9(03).
    02  MARITAL-W1                 PIC  X(01).
    02  ID-SPOUSE-W1               PIC  9(06).
    02  DEPT-CODE-W1               PIC  X(03).
    02  OFFICE-CODE-W1             PIC  X(03).
    02  MEMBER-TYPE-W1             PIC  X(01).
    02  L-QUEUE-W1                 PIC  9(02).
    02  AMT-LOAN-W1                PIC  9(05).
    02  L-STATUS-W1.
        03  EMERGENCY-W1           PIC  X(01).
        03  ORDINARY-W1            PIC  X(01).
        03  EXTRA-W1               PIC  X(01).
    02  AMT-GUA-W1                 PIC  9(05).
    02  GUA-LIMIT-USED-W1          PIC  9(05).
    02  GUA-MEN-STATUS-W1          PIC  9(01).
    02  GUA-MEN-RECORD-W1          PIC  9(01).
    02  SALARY-W1                  PIC  9(05).
    02  O-INCOME-W1                PIC  9(04).
    02  X-PRES-SW1                 PIC  9(04).
    02  O-PAYMENT-W1               PIC  9(5)V99.
    02  M-ADJUST-W1                PIC  9(3)V99.
    02  OUT-COND-W1                PIC  X(01).
    02  OUT-DATE-W1                PIC  X(08).
    02  UPDATE-DATE-SW1            PIC  X(08).
    02  BLANK-SW1                  PIC  X(29).
01  LOAN-REC-W1.
    02  ID-KEY-LW1.
        03  ID-MEMBER-LW1          PIC  9(06).
        03  L-TYPE-LW1             PIC  X(01).
```

```
           03  A-I-LW1                    PIC  X(01).
       02  CHECK-LW1.
           03  P-T-LW1                    PIC  X(01).
           03  REC-STATUS-LW1             PIC  X(01).
           03  F-L-LW1                    PIC  X(01).
       02  L-AGREE-NO-W1                  PIC  9(07).
       02  L-DATE-W1                      PIC  9(08).
       02  PRINCIPAL-LW1                  PIC  9(06).
       02  L-FLAG-W1                      PIC  X(01).
       02  PRINCE-F-L-PAID-W1             PIC  9(05).
       02  PRINC-MONTHLY-PAID-W1          PIC  9(05).
       02  DEBT-REMAIN-W1                 PIC  9(06).
       02  TOTAL-PERIOD-W1                PIC  9(03).
       02  PERIOD-W1                      PIC  9(03).
       02  INT-F-L-PAID-W1                PIC  9(04)V99.
       02  INT-PRES-PAID-W1               PIC  9(04)V99.
       02  INT-DECRE-W1                   PIC  9(03)V99.
       02  INT-RATE-W1                    PIC  9(02)V9.
       02  INTX-W1                        PIC  9(05)V99.
       02  GUA-MEN-KEY-LW1.
           03  ID-GUA-MEN-LW1             PIC  9(06).
           03  SEQUENCE-L2W1              PIC  9(01).
       02  GUA-LAND-KEY-LW1.
           03  ID-GUA-LAND-LW1            PIC  9(09).
           03  SEQUENCE-L3W1              PIC  9(01).
       02  GUA-LAND-NUMB-W1               PIC  9(01).
       02  GUA-SHARE-W1                   PIC  9(06).
       02  GUA-DEPOSI-W1                  PIC  9(06).
       02  UPDATE-DATE-LW1                PIC  X(08).
       02  BLANK-LW1                      PIC  X(08).
01   LOAN-REC-W2.
       02  ID-KEY-LW2.
           03  ID-MEMBER-LW2              PIC  9(06).
           03  L-TYPE-LW2                 PIC  X(01).
           03  A-I-LW2                    PIC  X(01).
       02  CHECK-LW2.
           03  P-T-LW2                    PIC  X(01).
           03  REC-STATUS-LW2             PIC  X(01).
           03  F-L-LW2                    PIC  X(01).
       02  L-AGREE-NO-W2                  PIC  9(07).
       02  L-DATE-W2                      PIC  9(08).
       02  PRINCIPAL-LW2                  PIC  9(06).
       02  L-FLAG-W2                      PIC  X(01).
       02  PRINC-F-L-PAID-W2              PIC  9(05).
       02  PRINC-MONTHLY-PAID-W2          PIC  9(05).
       02  DEBT-REMAIN-W2                 PIC  9(06).
       02  TOTAL-PERIOD-W2                PIC  9(03).
       02  PERIOD-W2                      PIC  9(03).
       02  INT-F-L-PAID-W2                PIC  9(04)V99.
       02  INT-PRES-PAID-W2               PIC  9(04)V99.
       02  INT-DECRE-W2                   PIC  9(03)V99.
       02  INT-RATE-W2                    PIC  9(02)V9.
       02  INTX-W2                        PIC  9(05)V99.
       02  GUA-MEN-KEY-LW2.
           03  ID-GUA-MEN-LW2             PIC  9(06).
           03  SEQUENCE-L2W2              PIC  9(01).
       02  GUA-LAND-KEY-LW2.
           03  ID-GUA-LAND-LW2            PIC  9(09).
           03  SEQUENCE-L3W2              PIC  9(01).
       02  GUA-LAND-NUMB-W2               PIC  9(01).
       02  GUA-SHARE-W2                   PIC  9(06).
       02  GUA-DEPOSIT-W2                 PIC  9(06).
       02  UPDATE-DATE-LW2                PIC  X(08).
       02  BLANK-LW2                      PIC  X(08).
01   PERS-GUA-REC-W1.
       02  GUA-MEN-KEY-PW1.
           03  ID-GUA-MEN-PW1             PIC  9(06).
           03  SEQUENCE-PW1               PIC  9(01).
           03  A-I-PW1                    PIC  X(01).
       02  CHECK-PW1.
           03  P-T-PW1                    PIC  X(01).
           03  REC-STATUS-PW1             PIC  X(01).
           03  F-L-PW1                    PIC  X(01).
       02  ID-GUAED-W1                    PIC  9(06).
       02  GUA-LIMIT-W1                   PIC  9(05).
       02  ID-CO-GUA-KEY-W1.
           03  ID-CO-GUA-W1               PIC  9(06).
           03  SEQUENCE-P2W1              PIC  9(01).
```

```
    02  UPDATE-DATE-PW1              PIC  X(08).
    02  BLANK-PW1                    PIC  X(27).
01  PERS-GUA-REC-W2.
    02  GUA-MEN-KEY-PW2.
        03  ID-GUA-MEN-PW2           PIC  9(06).
        03  SEQUENCE-PW2             PIC  9(01).
        03  A-I-PW2                  PIC  X(01).
    02  CHECK-PW2.
        03  P-T-PW2                  PIC  X(01).
        03  REC-STATUS-PW2           PIC  X(01).
        03  F-L-PW2                  PIC  X(01).
    02  ID-GUAED-W2                  PIC  9(06).
    02  GUA-LIMIT-W2                 PIC  9(05).
    02  ID-CO-GUA-KEY-W2.
        03  ID-CO-GUA-W2             PIC  9(06).
        03  SEQUENCE-P2W2            PIC  9(01).
    02  UPDATE-DATE-PW2              PIC  X(08).
    02  BLANK-PW2                    PIC  X(27).
01  ASSET-GUA-REC-W1.
    02  GUA-LAND-KEY-AW1.
        03  ID-GUA-LAND-AW1          PIC  9(09).
        03  SEQUENCE-A1W1            PIC  9(01).
        03  A-I-AW1                  PIC  X(01).
    02  CHECK-AW1.
        03  P-T-AW1                  PIC  X(01).
        03  REC-STATUS-AW1           PIC  X(01).
        03  F-L-AW1                  PIC  X(01).
    02  L-TYPE-AW1                   PIC  X(01).
    02  ID-MEMBER-AW1                PIC  9(06).
    02  PRINCIPAL-AW1                PIC  9(06).
    02  GUA-LIMIT-AW1                PIC  9(06).
    02  GUA-LAND-CO-KEY-W1.
        03  ID-GUA-LAND-CO-W1        PIC  9(09).
        03  SEQUENCE-A2W1            PIC  9(01).
    02  UPDATE-DATE-AW1              PIC  X(08).
    02  BLANK-AW1                    PIC  X(13).
01  LAND-STAT-REC-W1.
        02  GUA-LAND-KEY-DW1.
        03  ID-GUA-LAND-DW1          PIC  9(09).
        03  A-I-DW1                  PIC  X(01).
    02  CHECK-DW1.
        03  P-T-DW1                  PIC  X(01).
        03  REC-STATUS-DW1           PIC  X(01).
        03  F-L-DW1                  PIC  X(01).
    02  L-TYPE-DW1                   PIC  X(01).
    02  GUA-LAND-RECORD-W1           PIC  9(01).
    02  MEN-PER-LAND-W1              PIC  9(01).
    02  EVAL-DATE-W1                 PIC  9(08).
    02  LAND-AREA-W1                 PIC  9(04).
    02  EVAL-UNIT-W1                 PIC  9(06).
    02  EVAL-PRICE-W1                PIC  9(07).
    02  EVAL-BUILDING-W1             PIC  9(07).
    02  FIRE-INSURANCE-W1            PIC  9(06).
    02  AMT-GUA-SECUR-W1             PIC  9(06).
    02  REMAIN-SECUR-W1              PIC  9(06).
    02  UPDATE-DATE-DW1              PIC  X(08).
    02  BLANK-DW1                    PIC  X(54).
01  TRAN-REC.
    02  ID-KEY-TR.
        03  ORDER-NO.
            05  YEAR-TR              PIC  9(04).
            05  MONTH-TR             PIC  9(02).
            05  APPLY-NO             PIC  9(04).
        03  SEQUENCE-NO              PIC  9(01).
    02  IDTR                         PIC  9(06).
    02  L-TYPE-TR                    PIC  X(01).
    02  PRINCIPAL-TR                 PIC  9(06).
    02  TOTAL-PERIOD-TR              PIC  9(03).
    02  ID-GUA-TR                    PIC  9(09).
    02  NEW-GUA REDEFINES ID-GUA-TR.
        03  F-ID                     PIC  9(03).
        03  L-ID                     PIC  9(06).
    02  GUA-STAT-TR                  PIC  X(01).
    02  GUA-NUM-TR                   PIC  9(01).
    02  GUA-LIMIT-TR                 PIC  9(06).
    02  NEW-LIMIT REDEFINES GUA-LIMIT-TR.
        03  F-LIMIT                  PIC  9(01).
        03  L-LIMIT                  PIC  9(05).
    02  GUA-SHARE-TR                 PIC  9(06).
    02  GUA-DEPOSIT-TR               PIC  9(06).
    02  EVAL-DATE-TR                 PIC  9(08).
```

```
           02  LAND-AREA-TR              PIC  9(04).
           02  EVAL-UNIT-TR              PIC  9(06).
           02  EVAL-BUILDING-TR          PIC  9(07).
           02  FIRE-INSURANCE-TR         PIC  9(06).
           02  UPDATE-DATE-TR            PIC  9(08).
           02  BLANK-TR                  PIC  X(33).
       01  L-STAT-WK.
           02  EMERGENCY-L               PIC  X(01).
           02  ORDINARY-L                PIC  X(01).
           02  EXTRA-L                   PIC  X(01).
       01  L-IDGUA.
           02  ID-ZERO                   PIC  9(03).
           02  ID-GUAT                   PIC  9(06).
       01  DATE-CARD.
           02  DATE-YY                   PIC  9(04).
           02  DATE-MM                   PIC  9(02).
           02  DATE-DD                   PIC  9(02).
       01  KEEPRECM.
           02  K-MAN                     PIC  9(07).
           02  K-COM                     PIC  X(01).
       01  KEEPRECL.
           02  K-LAND                    PIC  9(10).
           02  K-COL                     PIC  X(01).
       01  HEAD-1.
           02  FILLER                    PIC  X(53)  VALUE SPACE.
           02  FILLER                    PIC  X(30)  VALUE 'PREPARE AND UP
      -        'DATE LOAN STATUS'.
           02  FILLER                    PIC  X(52)  VALUE SPACE.
       01  HEAD-TRAN.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(19)  VALUE '****TRANSACTIO
      -        'N****'.
           02  FILLER                    PIC  X(113)  VALUE SPACE.
       01  HEAD-STATID.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(21)  VALUE 'ID-MEMBER STAT
      -    'US FILE'.
           02  FILLER                    PIC  X(111) VALUE SPACE.
       01  HEAD-BEFORE.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(06)  VALUE 'BEFORE'.
           02  FILLER                    PIC  X(126) VALUE SPACE.
       01  HEAD-AFTER.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(05)  VALUE 'AFTER'.
           02  FILLER                    PIC  X(127) VALUE SPACE.
       01  HEAD-STATGUA.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(28)  VALUE 'GUARANTEE-PERS
      -    'ON STATUS FILE'.
           02  FILLER                    PIC  X(104) VALUE SPACE.
       01  HEAD-PERSID.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(21)  VALUE 'GUARANTEE-PERS
      -    'ON FILE'.
           02  FILLER                    PIC  X(111) VALUE SPACE.
       01  HEAD-ASSET.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(20)  VALUE 'ASSET-GUARANTE
      -    'E FILE'.
           02  FILLER                    PIC  X(112) VALUE SPACE.
       01  HEAD-LAND.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(16)  VALUE 'LAND-STATUS FI
      -    'LE'.
           02  FILLER                    PIC  X(16)  VALUE SPACE.
       01  HEAD-TEMP.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(19)  VALUE 'TEMPORARY LOAN
      -    ' FILE'.
           02  FILLER                    PIC  X(113) VALUE SPACE.
       01  HEAD-LOAN.
           02  FILLER                    PIC  X(01)  VALUE SPACE.
           02  FILLER                    PIC  X(09)  VALUE 'LOAN FILE'.
           02  FILLER                    PIC  X(123) VALUE SPACE.
       01  DATA-TRAN.
           02  FILLER                    PIC  X(05)  VALUE SPACE.
           02  DETAIL-T                  PIC  X(128).
```

```
 01  DATA-128.
     02  FILLER                      PIC  X(05)  VALUE SPACE.
     02  DETAIL-L                    PIC  X(123).
 01  DATA-64.
     02  FILLER                      PIC  X(05)  VALUE SPACE.
     02  DETAIL-S                    PIC  X(64).
     02  FILLER                      PIC  X(64) VALUE SPACE.
 01  LAST-LINE.
     02  FILLER                      PIC  X(48)  VALUE SPACE.
     02  FILLER                      PIC  X(32)  VALUE 'NUMBER OF UPDA
-    'TED TRANSACTION  ='.
     02  FILLER                      PIC  X(03)  VALUE SPACE.
     02  NO-TRAN                     PIC  ZZ9.
     02  FILLER                      PIC  X(47)  VALUE SPACE.
 01  HEAD-DUM.
     02  FILLER                      PIC  X(48)  VALUE SPACE.
     02  FILLER                      PIC  X(32)  VALUE 'NUMBER OF DUMM
-        'Y GUARANTOR       ='.
     02  FILLER                      PIC  X(03)  VALUE SPACE.
     02  NO-DUM                      PIC  ZZ9.
     02  FILLER                      PIC  X(47)  VALUE SPACE.
 01  HEAD-DATE.
     02  FILLER                      PIC  X(58)  VALUE SPACE.
     02  FILLER                      PIC  X(07)  VALUE 'DATE   '.
     02  H-DD                        PIC  9(02).
     02  FILLER                      PIC  X(01)  VALUE '/'.
     02  H-MM                        PIC  9(02).
     02  FILLER                      PIC  X(01)  VALUE '/'.
     02  H-YY                        PIC  9(04).
     02  FILLER                      PIC  X(38)  VALUE SPACE.
     02  FILLER                      PIC  X(07)  VALUE 'PAGE   '.
     02  H-PAGE                      PIC  ZZZ.
     02  FILLER                      PIC  X(10)  VALUE SPACE.
 01  HEAD-LINE.
     02  L1                          PIC  X(133).
 01  KEEP-RECORD.
     02  L2                          PIC  X(128)  VALUE SPACE.
 PROCEDURE DIVISION.
 ACCEPT-DATE.
     ACCEPT DATE-CARD.
     ADD 1 TO SW-PAGE.
 OPEN-FILE.
     OPEN I-O STAT-FILE.
     IF CHKS NOT = '00' GO TO CHKRTN-S.
     OPEN I-O LOAN-FILE.
     IF CHKL NOT = '00' GO TO CHKRTN-L.
     OPEN I-O PERS-GUA-FILE.
     IF CHKP NOT = '00' GO TO CHKRTN-P.
     OPEN I-O ASSET-GUA-FILE.
     IF CHKA NOT = '00' GO TO CHKRTN-A.
     OPEN I-O LAND-STAT-FILE.
     IF CHKD NOT = '00' GO TO CHKRTN-D.
     OPEN I-O TEMP-LOAN-FILE.
     IF CHKT NOT = '00' GO TO CHKRTN-T.
     OPEN INPUT DATA-FILE.
     IF CHKC NOT = '00' GO TO CHKRTNC.
     OPEN OUTPUT PRINT-FILE.
 PRINT-HEADING.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC AFTER POSITIONING 0.
     MOVE ALL '-' TO L1.
     WRITE PRINT-REC FROM HEAD-LINE AFTER POSITIONING 2.
     WRITE PRINT-REC FROM HEAD-1 AFTER POSITIONING 2.
     ADD 1 TO PAGE-NO.
     MOVE PAGE-NO TO H-PAGE.
     MOVE 0 TO LINE-C.
     MOVE DATE-DD TO H-DD.
     MOVE DATE-MM TO H-MM.
     MOVE DATE-YY TO H-YY.
     WRITE PRINT-REC FROM HEAD-DATE AFTER POSITIONING 2.
     MOVE ALL '-' TO L1.
     WRITE PRINT-REC FROM HEAD-LINE AFTER POSITIONING 2.
 PRINT-HEAD-EXIT.  EXIT.
 N-PAGE.
     ADD 1 TO LINE-C.
     IF LINE-C = 30 PERFORM PRINT-HEADING THRU PRINT-HEAD-EXIT.
 N-PAGE-EXIT.  EXIT.
 P-START.
     MOVE DATE-YY TO YEAR-TR.
     MOVE DATE-MM TO MONTH-TR.
     MOVE 1 TO APPLY-NO.
```

```
           MOVE 1 TO SEQUENCE-NO.
           MOVE ID-KEY-TR TO ID-TRAN-KEY.
           START DATA-FILE KEY = ID-TRAN-KEY.
       READ-TRAN.
           READ DATA-FILE NEXT RECORD INTO TRAN-REC AT END GO TO
               END-RTN.
           IF CHKC NOT = '00' GO TO CHKRTNC.
           IF ID-TRAN-KEY = 90000000000 GO TO END-RTN.
           IF ORDER-NO = K-ORDER NEXT SENTENCE
           ELSE IF SW-PAGE = 0
               PERFORM PRINT-HEADING THRU PRINT-HEAD-EXIT.
           MOVE 0 TO SW-PAGE.
           MOVE 0 TO SW-D.
           WRITE PRINT-REC FROM HEAD-TRAN AFTER POSITIONING 2.
           MOVE TRAN-REC TO DETAIL-T.
           WRITE PRINT-REC FROM DATA-TRAN AFTER POSITIONING 2.
           IF ID-GUA-TR NOT = 0 AND GUA-LIMIT-TR = 0
               DISPLAY '****THIS GUARANTOR DOES NOT HAVE GUA-LIMIT IN AP
      -        'PLICATION FORM'
               GO TO READ-AGAIN.
           IF ORDER-NO = K-ORDER GO TO CHECK-LANDSTATUS.
           MOVE 0 TO CX.
           ADD 1 TO COUNT-TRAN.
           MOVE ORDER-NO TO K-ORDER.
           MOVE IDTR TO ID-TRAN.
           MOVE ID-GUA-TR TO L-IDGUA.
      *****READ STAT-FILE ****
       READ-F21.
           MOVE SPACE TO PRINT-REC
           WRITE PRINT-REC FROM HEAD-STATID AFTER POSITIONING 2.
           WRITE PRINT-REC FROM HEAD-BEFORE AFTER POSITIONING 2.
       READ-S.
           MOVE ID-TRAN TO ID-MEMBER-S.
           MOVE '0' TO A-I-S.
           READ STAT-FILE INTO STAT-REC-W1
               INVALID KEY GO TO INVA-READSA.
           IF CHKS NOT = '00' GO TO CHKRTNS.
           MOVE STAT-REC-W1 TO DETAIL-L.
           PERFORM N-PAGE THRU N-PAGE-EXIT.
           WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
       READ-S-EXIT. EXIT.
       CHECK-A2.
           MOVE L-STATUS-W1 TO L-STAT-WK.
           IF CHECK-SW1 = '000' GO TO CHECK-LTYPETRAN.
           IF CHECK-SW1 = '110' GO TO CHECK-S110.
           PERFORM N-PAGE THRU N-PAGE-EXIT.
           DISPLAY '****ERROR RECORD, THERE IS NOT ACTIVE RECORD ID'
               ID-MEMBER-SW1.
       READ-AGAIN.
           READ DATA-FILE NEXT RECORD INTO TRAN-REC
               AT END GO TO END-RTN.
           IF CHKC NOT = '00' GO TO CHKRTNC.
           IF ID-TRAN-KEY = 90000000000 GO TO END-RTN.
           IF ORDER-NO = K-ORDER GO TO READ-AGAIN.
           PERFORM PRINT-HEADING THRU PRINT-HEAD-EXIT.
           MOVE SPACE TO PRINT-REC.
           WRITE PRINT-REC FROM HEAD-TRAN AFTER POSITIONING 2.
           MOVE TRAN-REC TO DETAIL-T.
           WRITE PRINT-REC FROM DATA-TRAN AFTER POSITIONING 2.
           ADD 1 TO COUNT-TRAN.
           MOVE 0 TO CX.
           MOVE 0 TO SW-D.
           MOVE IDTR TO ID-TRAN.
           MOVE ORDER-NO TO K-ORDER.
           MOVE ID-GUA-TR TO L-IDGUA.
           GO TO READ-F21.
      *****CHECK ATNF OF F2 ****
       CHECK-S110.
           MOVE ID-TRAN TO ID-MEMBER-S.
           MOVE '1' TO A-I-S.
           READ STAT-FILE INVALID KEY GO TO INVA-READSI.
           IF CHKS NOT = '00' GO TO CHKRTNS.
           MOVE STAT-REC TO DETAIL-L.
           PERFORM N-PAGE THRU N-PAGE-EXIT.
           WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
           PERFORM N-PAGE THRU N-PAGE-EXIT.
           IF CHECK-S = '100'
```

```
           IF GUA-MEN-STATUS   GUA-MEN-STATUS-W1 GO TO MAKE-NOF2
           ELSE DISPLAY '****NOT ALLOW TO LOAN BECAUSE HAVE TEMPORAR
-          'Y RECORD ID ' ID-TRAN.
           GO TO READ-AGAIN.
       DISPLAY '****ERROR RECORD, HAVE TEMPORARY RECORD BUT NO HAVE
-          'TO BE DELLTED RECORD ' ID-TRAN.
       GO TO READ-AGAIN.
*****MAKE NO PERSON WHO IS GUARANTOR AND WANT TO LOAN****
 MAKE-NOF2.
       DISPLAY '****LET THIS ID-MEMBER LOAN BEFORE TO BE GUARANTOR,
-      'ID-MEMBER ' ID-TRAN.
       DISPLAY '****WAIT FOR LOAN BECAUSE OF DUMMY GUARANTOR, ID-MEM
-      'BER ' ID-TRAN.
       PERFORM USE-DUMMYID THRU USE-DUMMYID-EXIT.
       MOVE ID-TRAN TO ID-GUA-MEN-P.
       MOVE GUA-MEN-RECORD TO SEQUENCE-P1.
       MOVE '0' TO A-I-P.
 P-READ5.
       READ PERS-GUA-FILE INVALID KEY GO TO INVA-READPA.
       IF CHKP NOT = '00' GO TO CHKRTNP.
       IF CHECK-P = '110' NEXT SENTENCE
       ELSE DISPLAY '**** ERROR, HAS NO TEMPORARY RECORD OF GUARANTE
-      'E PERSON FILE ****' ID-TRAN.
 P-READ5-EXIT.  EXIT.
 P-READ4.
       MOVE ID-GUAED TO ID-MEMBER-L.
       MOVE '2' TO L-TYPE-L.
       MOVE '0' TO A-I-L1.
       READ LOAN-FILE INVALID KEY GO TO INVA-READLA.
       IF CHKL NOT = '00'  GO TO CHKRTNL.
       IF CHECK-L = '110' NEXT SENTENCE
       ELSE DISPLAY '**** ERROR HAS NO TEMPORARY RECORD OF LOAN-FILE
-      ' ****' ID-MEMBER-L.
       IF ID-GUA-MEN-L NOT = ID-TRAN GO TO P-DF5.
       MOVE DUM-ID TO ID-GUA-MEN-L.
       REWRITE LOAN-REC INVALID KEY GO TO INVA-REWRITELA.
 P-DF5.
       MOVE '111' TO CHECK-P.
       REWRITE PERS-GUA-REC INVALID KEY GO TO IN-REP.
       IF CHKP NOT = '00'  GO TO CHKRTNP.
       MOVE '1' TO A-I-P.
       MOVE '010' TO CHECK-P.
       WRITE PERS-GUA-REC INVALID KEY GO TO INVA-WRITEPA.
       IF CHKP NOT = '00'  GO TO CHKRTNP.
 P-DF5-EXIT.  EXIT.
 P-DUM1.
       MOVE DUM-ID TO ID-GUA-MEN-P.
 P-DUM2.
       MOVE '0' TO A-I-P.
       MOVE '110' TO CHECK-P.
       WRITE PERS-GUA-REC INVALID KEY GO TO INVA-WRITEPA.
       IF CHKP NOT = '00'  GO TO CHKRTNP.
 P-DUM2-EXIT.  EXIT.
 P-OLDSTATUS.
       DISPLAY '****CHANGE TO OLD STATUS'.
       MOVE '0' TO A-I-S.
       MOVE '000' TO CHECK-S.
       REWRITE STAT-REC INVALID KEY GO TO INVA-REWRITESA.
       IF CHKS NOT = '00' GO TO CHKRTNS.
       MOVE STAT-REC TO DETAIL-L.
       PERFORM N-PAGE THRU N-PAGE-EXIT.
       WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
       MOVE '1' TO A-I-SW1.
       MOVE '010' TO CHECK-SW1.
       REWRITE STAT-REC FROM STAT-REC-W1
           INVALID KEY GO TO INVA-REWRITESI.
       IF CHKS NOT = '00' GO TO CHKRTNS.
       MOVE STAT-REC-W1 TO DETAIL-L.
       PERFORM N-PAGE THRU N-PAGE-EXIT.
       WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
       GO TO READ-F21.
```

```
 P-SF5.
     MOVE ID-GUA-MEN-L TO ID-GUA-MEN-P.
     MOVE SEQUENCE-L2 TO SEQUENCE-P1.
 P-SF51.
     MOVE '0' TO A-I-P.
     PERFORM P-READ5 THRU P-READ5-EXIT.
     IF ID-CO-GUA = ID-TRAN
        PERFORM P-DF5 THRU P-DF5-EXIT
        MOVE DUM-ID TO ID-CO-GUA
        GO TO P-DUM2.
     MOVE ID-CO-GUA TO ID-GUA-MEN-P.
     MOVE SEQUENCE-P2 TO SEQUENCE-P1.
     GO TO P-SF51.
*****CHECK NEW LOAN TYPE ****
 CHECK-LTYPETRAN.
     IF L-TYPE-TR = '1' GO TO NEW-EMERGENCY.
     IF L-TYPE-TR = '2' GO TO NEW-ORDINARY.
     IF L-TYPE-TR = '3' OR '4' NEXT SENTENCE
     ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
     DISPLAY '****LOAN-TYPE CODE ERROR FROM TRANSACTION-ID'
        ID-TRAN
     GO TO READ-AGAIN.
     IF EMERGENCY-L = 1
     PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT ALLOW TO HAVE EXTRA LOAN BECAUSE HAVE EM
-       'ERGENCY LOAN, TRANSACTION-ID' ID-TRAN
        GO TO READ-AGAIN.
     IF L-QUEUE = 0 NEXT SENTENCE
     ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
     DISPLAY '****NOT ALLOW TO LOAN BECAUSE IT IS NOT DUE TIME TO
-    'LOAN, TRANSACTION-ID' ID-TRAN
     GO TO READ-AGAIN.
**********************************************************************
*****UPDATING STAT-FILE TO ACTIVE-TEMPORARY RECORD FOR EXTRA LOAN ****
     MOVE STAT-REC-W1 TO STAT-REC.
     MOVE '110' TO CHECK-S.
     IF L-TYPE-TR = '3' MOVE 10 TO L-QUEUE
     ELSE COMPUTE L-QUEUE = TOTAL-PERIOD-TR / 2.
     MOVE '0' TO EMERGENCY  ORDINARY.
     MOVE L-TYPE-TR TO EXTRA.
 REWRITE-STATREC.
     MOVE DATE-CARD TO UPDATE-DATE-S.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
     REWRITE STAT-REC INVALID KEY GO TO INVA-REWRITESA.
     IF CHKS NOT = '00' GO TO CHKRTNS.
     MOVE STAT-REC TO L2.
*****CHANGE APOF TO ITOF ****
 CHANGE-TOITOF2.
     MOVE '1' TO A-I-SW1.
     MOVE '100' TO CHECK-SW1.
     MOVE DATE-CARD TO UPDATE-DATE-SW1.
*****READ STAT-FILE BY IDI FOR UPDATING****
     MOVE '1' TO A-I-S.
     READ STAT-FILE INVALID KEY GO TO CHECK-S23.
     IF CHKS NOT = '00' GO TO CHKRTNS.
     REWRITE STAT-REC FROM STAT-REC-W1
        INVALID KEY GO TO INVA-REWRITESI.
     IF CHKS NOT = '00' GO TO CHKRTNS.
     MOVE STAT-REC-W1 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
 PIM-ATNF.
     MOVE L2 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
 CHANGE-TOITOF2-EXIT.  EXIT.
 END-DETAILF2.
     GO TO CHECK-LANDSTATUS.
**********************************************************************
*****WANT TO HAVE EMERGENCY LOAN****
**********************************************************************

 NEW-EMERGENCY.
     IF EMERGENCY-L = '1' OR ORDINARY-L = '2' NEXT SENTENCE
     ELSE IF EXTRA-L = '3' OR '4'
     PERFORM N-PAGE THRU N-PAGE-EXIT
```

```
           DISPLAY '****NOT ALLOW TO HAVE EMERGENCY LOAN BECAUSE HAV
-          'E EXTRA LOAN, ID-MEMBER' ID-TRAN
           GO TO READ-AGAIN.
*****UPDATING STAT-FILE TO ACTIVE-TEMPORARY RECORD FOR EMERGENCY LOAN****
      MOVE STAT-REC-W1 TO STAT-REC.
      MOVE '110' TO CHECK-S.
      MOVE '1'   TO EMERGENCY.
      PERFORM REWRITE-STATREC THRU CHANGE-TOITOF2-EXIT.
*************************************************************************
*****READ TEMPORARY FILE****
 READ-TEMPFILE.
      MOVE IDTR TO ID-MEMBER-T.
      MOVE L-TYPE-TR TO L-TYPE-T.
      MOVE '0' TO A-I-T1.
      WRITE PRINT-REC FROM HEAD-TEMP AFTER POSITIONING 2.
      READ TEMP-LOAN-FILE INVALID KEY GO TO INVA-READTA.
      IF CHKT NOT = '00' GO TO CHKRTNT.
      IF CHECK-T = '110' NEXT SENTENCE
      ELSE DISPLAY '****THIS RECORD HAS NO ATNF OF TEMP-FILE***'
           ID-KEY-T
           GO TO READ-AGAIN.
      MOVE TEMP-LOAN-REC TO DETAIL-T.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
 READ-TEMPFILE-EXIT.  EXIT.
*****REWRITE TEMPORRARY FILE TO LOAN FILE****
 UPDATE-LOANFILE.
      MOVE IDTR TO ID-MEMBER-L.
      MOVE L-TYPE-TR TO L-TYPE-L.
      MOVE '0' TO A-I-L1.
      MOVE SPACE TO PRINT-REC.
      WRITE PRINT-REC FROM HEAD-LOAN AFTER POSITIONING 2.
      WRITE PRINT-REC FROM HEAD-BEFORE AFTER POSITIONING 1.
      READ LOAN-FILE INTO LOAN-REC-W1.
      IF CHKL = '23' GO TO WRITE-LOANFILE.
      IF CHKL NOT = '00' GO TO CHKRTNL.
      MOVE LOAN-REC-W1 TO DETAIL-L.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
      IF CHECK-LW1 = '000' OR '001' GO TO REWRITE-LOAN-FILE.
      IF CHECK-LW1 = '110'
      PERFORM N-PAGE THRU N-PAGE-EXIT
           DISPLAY '****NOT CHANGE BECAUSE HAVE TEMPORARY RECORD ID'
-           IDTR
           GO TO READ-AGAIN.
*****WRITE LOAN FILE1****
 WRITE-LOANFILE.
      DISPLAY '****NOT HAVE SAME LOAN IN OLD DEBT****'   IDTR.
      WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
      MOVE '110' TO CHECK-T.
      MOVE GUA-SHARE-TR TO GUA-SHARE-T.
      IF L-TYPE-T = 2 MOVE K-MAN TO GUA-MEN-KEY-T.
      IF L-TYPE-T = 3 OR 4 MOVE K-LAND TO GUA-LAND-KEY-T.
      MOVE DATE-CARD TO UPDATE-DATE-T.
      MOVE TEMP-LOAN-REC TO LOAN-REC.
      WRITE LOAN-REC
           INVALID KEY GO TO INVA-WRITELA.
      IF CHKL NOT = '00' GO TO CHKRTNL.
      MOVE TEMP-LOAN-REC TO DETAIL-L.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
      GO TO READ-TRAN.
*****REWRITE LOAN FILE*****
 REWRITE-LOAN-FILE.
      MOVE '110' TO CHECK-T.
      MOVE GUA-SHARE-TR TO GUA-SHARE-T.
      MOVE GUA-DEPOSIT-TR TO GUA-DEPOSIT-T.
      IF L-TYPE-T = 2 MOVE K-MAN TO GUA-MEN-KEY-T.
      IF L-TYPE-T = 3 OR 4 MOVE K-LAND TO GUA-LAND-KEY-T.
      MOVE DATE-CARD TO UPDATE-DATE-T.
      REWRITE LOAN-REC FROM TEMP-LOAN-REC
           INVALID KEY GO TO INVA-REWRITELA.
      IF CHKL NOT = '00' GO TO CHKRTNL.
      WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
      IF CHECK-LW1 = '001'
           MOVE TEMP-LOAN-REC TO DETAIL-L
           PERFORM N-PAGE THRU N-PAGE-EXIT
           WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1
           GO TO READ-TRAN.
```

```
     MOVE SPACE TO L2.
     MOVE TEMP-LOAN-REC TO L2.
 UPDATE-TOLI.
*****READ LOAN-FILE BY IDI FOR UPDATING****
     MOVE '1' TO A-I-LW1.
     MOVE '100' TO CHECK-LW1.
     MOVE DATE-CARD TO UPDATE-DATE-LW1.
     MOVE IDTR TO ID-MEMBER-L.
     MOVE L-TYPE-TR TO L-TYPE-L.
     MOVE '1' TO A-I-LI.
     READ LOAN-FILE INVALID KEY GO TO CHECK-L23.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     REWRITE LOAN-REC FROM LOAN-REC-W1
         INVALID KEY GO TO INVA-REWRITELI.
     IF CHKL NO1 = '00' GO TO CHKRTNL.
     MOVE LOAN-REC-W1 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
 PIM-LOAN.
     MOVE L2 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
     GO TO READ-TRAN.
**********************************************************************
*****WANT TO HAVE ORDINARY LOAN****
**********************************************************************

 NEW-ORDINARY.
     IF L-QUEUE = 0 NEXT SENTENCE
     ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
     DISPLAY '****NOT ALLOW TO LOAN BECAUSE IT IS NOT DUE TIME TO
-    'LOAN, TRANSACTION-ID' ID-TRAN
     GO TO READ-AGAIN.
*****UPDATING STAT-FILE TO ACTIVE-TEMPORARY RECORD FOR ORDINARY LOAN****
     MOVE STAT-REC-W1 TO STAT-REC.
     MOVE '110' TO CHECK-S.
     MOVE 10 TO L-QUEUE.
     MOVE '2' TO ORDINARY.
     MOVE '0' TO EXTRA.
     PERFORM REWRITE-STATREC THRU CHANGE-TOITOF2-EXIT.
 NEW-ORDINARY-EXIT. EXIT.
**********************************************************************
*****CHECK LAND-STATUS AFTER CHECK F2****
 CHECK-LANDSTATUS.
     IF L-TYPE-TR = '2' GO TO CHECK-GUAMAN.
     IF EXTRA-L = '3' GO TO EXTRA-3.
     IF EXTRA-L = '4' GO TO EXTRA-4.
     IF L-TYPE-TR = '3' OR '4' GO TO EXTRA-3.
**********************************************************************
*****CHECK FILE STATUS GUARANTOR****
**********************************************************************

 CHECK-GUAMAN.
     MOVE 0 TO SW-D.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-STATGUA AFTER POSITIONING 2.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-BEFORE AFTER POSITIONING 2.
     IF ID-GUA-TR = 0 AND GUA-SHARE-TR = 0 AND GUA-DEPOSIT-TR = 0
         DISPLAY '****USE DUMMY GUARANTOR****'
         GO TO USE-DUMMYID.
     IF L-ID = 0 MOVE ZERO TO GUA-MEN-KEY-T
         GO TO READ-TAGAIN.
     MOVE L-ID TO ID-MEMBER-S.
     MOVE '0' TO A-I-S.
     READ STAT-FILE INTO STAT-REC-W1
         INVALID KEY GO TO INVA-READSA.
     IF CHKS NOT = '00' GO TO CHKRTNS.
     MOVE STAT-REC-W1 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
     IF CHECK-S = '000' GO TO CHANGE-RECGUA.
     IF CHECK-S = '110' NEXT SENTENCE
     ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
     DISPLAY '****ERROR RECORD, THERE IS NOT ACTIVE RECORD ID ' ID
-    -GUA-TR
     GO TO READ-AGAIN.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****NOT CHANGE TEMPORARY RECORD GUARANTEE-ID ' L-ID.
```

```
USE-DUMMYID.
     MOVE '90000010' TO GUA-MEN-KEY-P.
     READ PERS-GUA-FILE INVALID KEY GO TO INVA-READPA.
     IF CHKP NOT = '00' GO TO CHKRTNP.
     MOVE ID-GUAED TO DUM-ID.
     COMPUTE ID-GUAED = ID-GUAED - 1.
     MOVE DATE-CARD TO UPDATE-DATE-P.
     REWRITE PERS-GUA-REC INVALID KEY GO TO INVA-REWRITEPA.
     IF CHKP NOT = '00' GO TO CHKRTNP.
USE-DUMMYID-EXIT.  EXIT.
P-USE-D.
*****CREATE DUMMY-ID IN F5****
     ADD 1 TO SW-D.
     ADD 1 TO DUM.
     MOVE DUM-ID TO ID-GUA-MEN-P.
     MOVE 1 TO SEQUENCE-P1.
     MOVE '0' TO A-I-P.
     READ PERS-GUA-FILE INVALID KEY GO TO CHECK-P23.
     IF CHKP NOT = '00' GO TO CHKRTNP.
     MOVE '1' TO SW-1.
     GO TO P-KEYF5.
UPDATE-LOANFILE2.
     IF EXTRA-L = '3' OR '4' GO TO CHANGE-3I.
     GO TO UPDATE-LOANFILE.
CHANGE-3I.
     MOVE IDTR TO ID-MEMBER-L.
     MOVE EXTRA-L TO L-TYPE-L.
     MOVE '0' TO A-I-L1.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-LOAN AFTER POSITIONING 2.
     WRITE PRINT-REC FROM HEAD-BEFORE AFTER POSITIONING 2.
     READ LOAN-FILE INTO LOAN-REC-W2
         INVALID KEY GO TO INVA-READLA.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     MOVE LOAN-REC-W2 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
*****CHANGE APOF TO ITOF OF F4****
CHANGE-TOITOF4.
     MOVE LOAN-REC-W2 TO LOAN-REC-W1.
     MOVE '1' TO A-I-LW1.
     MOVE '100' TO CHECK-LW1.
     MOVE DATE-CARD TO UPDATE-DATE-LW1.
     MOVE ID-KEY-LW1 TO ID-KEY-L.
     READ LOAN-FILE INVALID KEY GO TO CHECK-L232.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     REWRITE LOAN-REC FROM LOAN-REC-W1
         INVALID KEY GO TO INVA-REWRITELI.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     MOVE LOAN-REC-W1 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
CHANGE-TOAPOL4.
     MOVE '1' TO F-L-LW2.
     MOVE DATE-CARD TO UPDATE-DATE-LW2.
     REWRITE LOAN-REC FROM LOAN-REC-W2.
     IF CHKL = '23' WRITE LOAN-REC FROM LOAN-REC-W2.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     MOVE LOAN-REC-W2 TO DETAIL-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
     MOVE 0 TO CX.
     MOVE IDTR TO ID-MEMBER-L.
     MOVE L-TYPE-TR TO L-TYPE-L.
     MOVE '0' TO A-I-L1.
     READ LOAN-FILE.
     IF CHKL = '23' GO TO WRITE-LOANFILE.
     IF CHKL NOT = '00' GO TO CHKRTNL.
     MOVE CHECK-L TO CHECK-LW1.
     GO TO REWRITE-LOAN-FILE.
P-CALGUAUSE.
     MOVE ID-MEMBER-SW1 TO ID-GUA-MEN-P.
     ADD 1 TO SEQUENCE-P1.
     MOVE '0' TO A-I-P.
     READ PERS-GUA-FILE.
```

```
    IF CHKP = '23'  GO TO P-CALGUAUSE.
    IF CHKP NOT = '00'  GO TO CHKKTNP.
    IF CHECK-P = '000' NEXT SENTENCE
    ELSE  GO TO P-CALGUAUSE.
    IF IDTR = ID-GUAED NEXT SENTENCE
    ELSE  GO TO P-CALGUAUSE.
    COMPUTE CAL-GUA-USED = GUA-LIMIT-USED-W1 - GUA-LIMIT.
    COMPUTE C-REMAIN = AMT-GUA-W1 - CAL-GUA-USED.
P-CALGUAUSE-EXIT.   EXIT.
CHANGE-RECGUA.
    MOVE 0 TO C-REMAIN CAL-GUA-USEQ CAL-MEN CAL-RECORD.
    IF GUA-STAT-TR = '0'
        MOVE 0 TO SEQUENCE-P1
        PERFORM P-CALGUAUSE THRU P-CALGUAUSE-EXIT
        GO TO P-CHECKREMAIN.
    COMPUTE C-REMAIN = AMT-GUA-W1 - GUA-LIMIT-USED-W1.
P-CHECKREMAIN.
    IF GUA-LIMIT-TR NOT   C-REMAIN NEXT SENTENCE
    ELSE
    PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****FIND NEW GUARANTOR BECAUSE AMOUNT-GUARANTEE
-       'IS NOT ENOUGH' ID-GUA-TR
        GO TO USE-DUMMYID.
    IF GUA-STAT-TR = '0'
        COMPUTE CAL-GUA-USED = C-REMAIN + L-LIMIT
        MOVE GUA-MEN-STATUS TO CAL-MEN
    ELSE
    COMPUTE CAL-GUA-USED = GUA-LIMIT-USED + L-LIMIT
    COMPUTE CAL-MEN = GUA-MEN-STATUS + 1.
    IF GUA-MEN-RECORD-W1   8
        ADD 1 TO CAL-RECORD
        GO TO READ-P.
    COMPUTE CAL-RECORD = GUA-MEN-RECORD-W1 + 1.
READ-P.
    MOVE ID-MEMBER-SW1 TO ID-GUA-MEN-P.
    MOVE CAL-RECORD TO SEQUENCE-P1.
    MOVE '0' TO A-I-P.
    READ PERS-GUA-FILE INTO PERS-GUA-REC-W2.
    IF CHKP = '23'
        MOVE '0' TO SW-1
        GO TO P-MOVESTAT.
    IF CHKP NOT = '00'
        DISPLAY 'READ PERS-GUA-FILE ERROR. STATUS KEY VALUE IS '
-       CHKP
        GO TO READ-AGAIN.
    IF CHECK-PW2 = '000' OR '110'
        ADD 1 TO CAL-RECORD
        GO TO READ-P.
    MOVE ID-MEMBER-SW1 TO ID-GUA-MEN-P.
    MOVE CAL-RECORD TO SEQUENCE-P1.
    MOVE '1' TO A-I-P.
    READ PERS-GUA-FILE INVALID KEY GO TO INVA-READPA.
    IF CHKP NOT = '00'
    PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY 'PERS-GUA-RECORD ERROR.NOT FOUND RECORD IPOF '
-       ID-GUA-MEN-P
        GO TO READ-AGAIN.
    MOVE '1' TO SW-1.
P-MOVESTAT.
    MOVE STAT-REC-W1 TO STAT-REC.
    MOVE '110' TO CHECK-S.
    MOVE CAL-GUA-USED TO GUA-LIMIT-USED.
    MOVE CAL-MEN TO GUA-MEN-STATUS.
    MOVE CAL-RECORD TO GUA-MEN-RECORD.
    PERFORM REWRITE-STATREC THRU CHANGE-TOITOF2-EXIT.
P-MOVESTAT-EXIT. EXIT.
**************************************************************
*****CREATE PERS-GUA FILE****
**************************************************************

P-KEYF5.
    IF CX > 4
        DISPLAY '****ERROR, USE GUARANTOR GREATER THAN GUA-MEN-ST
-       'ATUS ' ID-TRAN
        GO TO READ-AGAIN.
    ADD 1 TO CX.
```

```
    IF CX NOT   1 GO TO PREPARE-F5.
    IF SW-D NOT = 0 MOVE DUM-ID TO ID-CO-GUA-W1
                    MOVE 1 TO SEQUENCE-P2W1
                    PERFORM WRITE-F5 THRU WRITE-F5-EXIT
                    GO TO PREPARE-F5.
    MOVE L-ID TO ID-CO-GUA-W1.
    MOVE CAL-RECORD TO SEQUENCE-P2W1.
    PERFORM WRITE-F5 THRU WRITE-F5-EXIT.
PREPARE-F5.
    MOVE SW-1 TO SW-2.
    IF SW-D NOT = 0 MOVE DUM-ID TO ID-GUA-MEN-PW1
                    MOVE 1 TO SEQUENCE-PW1
    ELSE MOVE L-ID TO ID-GUA-MEN-PW1
    MOVE CAL-RECORD TO SEQUENCE-PW1 C-SEQUENCE.
    MOVE '0' TO A-I-PW1.
    IF CX = 1 MOVE GUA-MEN-KEY-PW1 TO KEEPRECM.
    MOVE '110' TO CHECK-PW1.
    MOVE ID-TRAN TO ID-GUAED-W1.
    MOVE L-LIMIT TO GUA-LIMIT-W1.
    MOVE ZEROES TO ID-CO-GUA-KEY-W1.
    MOVE DATE-CARD TO UPDATE-DATE-PW1.
    MOVE SPACE TO BLANK-PW1.
    IF GUA-NUM-TR = SEQUENCE-NO NEXT SENTENCE
    ELSE GO TO READ-TRAN.
WRITE-F5.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC FROM HEAD-PERSID AFTER POSITIONING 2.
    IF SW-2 = '1' THEN PERFORM 5-REWRITE THRU 5-REWRITE-EXIT
    ELSE PERFORM 5-WRITE THRU 5-WRITE-EXIT.
P-MOVEPER.
    MOVE PERS-GUA-REC-W1 TO DETAIL-S.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-64 AFTER POSITIONING 1.
WRITE-F5-EXIT.  EXIT.
READ-TAGAIN.
    PERFORM READ-TEMPFILE THRU READ-TEMPFILE-EXIT.
    GO TO UPDATE-LOANFILE2.
5-REWRITE.
    REWRITE PERS-GUA-REC FROM PERS-GUA-REC-W1
        INVALID KEY GO TO INVA-REWRITEPA.
    IF CHKP NOT = '00' GO TO CHKRTNPR.
5-REWRITE-EXIT.
5-WRITE.
    WRITE PERS-GUA-REC FROM PERS-GUA-REC-W1
        INVALID KEY GO TO INVA-WRITEPA.
    IF CHKP NOT = '00' GO TO CHKRTNPW.
5-WRITE-EXIT.
EXTRA-4.
    PERFORM EXTRA-3 THRU EXTRA-3-EXIT.
    IF CHECK-DW1 = '000' GO TO CHECK-OLDLAND4.
    IF CHECK-DW1 = '110'
    PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT CHANGE BECAUSE OF TEMPORARY RECORD ID***
-    '*' L-IDGUA
        GO TO DUMMY-LAND.
    GO TO PREPARE-F7N.
CHECK-OLDLAND4.
    IF GUA-STAT-TR = '1' GO TO USE-OLDLAND.
    IF L-TYPE-TR = '4' GO TO CAL-REMAIN.
    GO TO PREPARE-F7O.
EXTRA-3.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC FROM HEAD-LAND AFTER POSITIONING 2.
    IF ID-GUA-TR = 0 AND GUA-SHARE-TR = 0 AND GUA-DEPOSIT-TR = 0
        DISPLAY '****USE DUMMY LAND****'
        GO TO DUMMY-LAND.
    WRITE PRINT-REC FROM HEAD-BEFORE AFTER POSITIONING 2.
    MOVE ID-GUA-TR TO ID-GUA-LAND-D.
    MOVE '0' TO A-I-D.
    READ LAND-STAT-FILE INTO LAND-STAT-REC-W1
        INVALID KEY GO TO INVA-READDA.
    IF CHKD NOT = '00' GO TO CHKRTND.
    MOVE LAND-STAT-REC-W1 TO DETAIL-L.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
EXTRA-3-EXIT.  EXIT.
```

```
CHECK-FORMULA.
    IF  CHECK-DW1 = '000' GO TO CHECK-OLDLAND.
    IF  CHECK-DW1 = '110'
    PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT CHANGE BECAUSE OF TEMPORARY RECORD ID '
-       ' ID-GUA-TR
        GO TO DUMMY-LAND.
*****NOT FOUND IDAPOF OR ATNF*****
*****************************************************************
****PREPARE NEW LAND****
*****************************************************************

 PREPARE-F7N.
    DISPLAY '****NEVER USE THIS LAND****' ID-GUA-TR.
    WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
    MOVE ID-GUA-TR TO ID-GUA-LAND-DW1.
    MOVE '0' TO A-I-DW1.
    MOVE '110' TO CHECK-DW1.
    MOVE L-TYPE-TR TO L-TYPE-DW1.
    MOVE 1 TO GUA-LAND-RECORD-W1 MEN-PER-LAND-W1.
    MOVE EVAL-DATE-TR TO EVAL-DATE-W1.
    MOVE LAND-AREA-TR TO LAND-AREA-W1.
    MOVE EVAL-UNIT-TR TO EVAL-UNIT-W1.
    COMPUTE EVAL-PRICE-W1 = LAND-AREA-TR * EVAL-UNIT-TR.
    MOVE EVAL-BUILDING-TR TO EVAL-BUILDING-W1.
    MOVE FIRE-INSURANCE-TR TO FIRE-INSURANCE-W1.
    IF L-TYPE-TR = '3'
        COMPUTE AMT-GUA-SECUR-W1 = (EVAL-PRICE-W1 + EVAL-BUILDING
-       ) * 0.5.
    COMPUTE
    AMT-GUA-SECUR-W1 = (EVAL-PRICE-W1 + EVAL-BUILDING) * 0.6.
    COMPUTE REMAIN-SECUR-W1 = (AMT-GUA-SECUR-W1 - GUA-LIMIT-TR).
    MOVE DATE-CARD TO UPDATE-DATE-DW1.
    MOVE SPACE TO BLANK-DW1.
    MOVE LAND-STAT-REC-W1 TO LAND-STAT-REC.
    MOVE 1 TO CAL-RECORD.
 PREPARE-F7N-EXIT. EXIT.
 REWRITE-F71.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
    REWRITE LAND-STAT-REC FROM LAND-STAT-REC-W1
        INVALID KEY GO TO INVA-REWRITEDA.
    IF CHKD NOT = '00' GO TO CHKRTND.
    MOVE LAND-STAT-REC-W1 TO DETAIL-L.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
    GO TO CREATE-ASSET.
 CHECK-OLDLAND.
    IF GUA-STAT-TR = '0' GO TO CHECK-NEWEX3.
 USE-OLDLAND.
    IF GUA-LIMIT-TR   REMAIN-SECUR-W1
    PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT ALLOW TO LOAN BECAUSE REMAIN-SECURITY IS
-       ' NOT ENOUGH ID-MEMBER ' ID-TRAN
        GO TO READ-AGAIN.
    IF L-TYPE-DW1 = '3' GO TO CHECK-NEWEX3.
 CHECK-NEWEX4.
    IF L-TYPE-DW1 = '4' NEXT SENTENCE
    ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT ALLOW TO LOAN, MUST DIVIDE LAND BEFORE,
-       'ID-GUA-LAND' L-IDGUA
        GO TO READ-AGAIN.
    MOVE LAND-STAT-REC-W1 TO LAND-STAT-REC.
    MOVE L-TYPE-TR TO L-TYPE-D.
    MOVE '110' TO CHECK-D.
    GO TO CAL-REMAIN.
 CHECK-NEWEX3.
    IF L-TYPE-TR = '3' NEXT SENTENCE
    ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****NOT ALLOW TO LOAN BY USE LAND-ID' L-IDGUA
        GO TO READ-AGAIN.
*****************************************************************
****UPDATE OLD LAND****
*****************************************************************
 PREPARE-F70.
    MOVE 0 TO C-REMAIN CAL-AMTGUA TCAL-AMTGUA CAL-MEN CAL-RECORD.
    MOVE ID-GUA-TR TO ID-GUA-LAND-D.
```

```
    MOVE '0' TO A-I-D.
    MOVE '110' TO CHECK-D.
    MOVE L-TYPE-TR TO L-TYPE-D.
    MOVE EVAL-DATE-TR TO EVAL-DATE.
    MOVE LAND-AREA-TR TO LAND-AREA.
    MOVE EVAL-UNIT-TR TO EVAL-UNIT.
    COMPUTE EVAL-PRICE = (LAND-AREA * EVAL-UNIT).
    MOVE EVAL-BUILDING-TR TO EVAL-BUILDING.
    MOVE FIRE-INSURANCE-TR TO FIRE-INSURANCE.
    COMPUTE AMT-GUA-SECUR = (EVAL-PRICE + EVAL-BUILDING) * 0.5.
PREPARE-F70-EXIT.  EXIT.
CAL-REMAIN.
    IF GUA-STAT-TR = '1'  GO TO P-CAL2.
    MOVE 0 TO SEQUENCE-A1.
P-CAL1.
    MOVE ID-GUA-LAND-D TO ID-GUA-LAND-A.
    ADD 1 TO SEQUENCE-A1.
    MOVE '0' TO A-I-A1.
    READ ASSET-GUA-FILE.
    IF CHKA = '23'  GO TO P-CAL1.
    IF CHKA NOT = '00'  GO TO CHKRTNA.
    IF CHECK-A = '000' NEXT SENTENCE
    ELSE GO TO P-CAL1.
    IF ID-MEMBER-A = IDTR NEXT SENTENCE
    ELSE  GO TO P-CAL1.
    COMPUTE CAL-AMTGUA = (REMAIN-SECUR + GUA-LIMIT-A).
    COMPUTE TCAL-AMTGUA = (CAL-AMTGUA - GUA-LIMIT-TR).
    GO TO P-CAL3.
P-CAL2.
    COMPUTE CAL-AMTGUA = (AMT-GUA-SECUR - REMAIN-SECUR).
    COMPUTE TCAL-AMTGUA = (CAL-AMTGUA + GUA-LIMIT-TR).
    COMPUTE REMAIN-SECUR = (AMT-GUA-SECUR - TCAL-AMTGUA).
P-CAL3.
    IF GUA-LAND-RECORD  8 NEXT SENTENCE
    ELSE PERFORM N-PAGE THRU N-PAGE-EXIT
        DISPLAY '****PLEASE REORGANIZE RECORD SEQUENCE LAND-ID '
-   L-IDGUA.
    COMPUTE CAL-RECORD = (GUA-LAND-RECORD + 1).
    IF GUA-STAT-TR = '0'
        MOVE MEN-PER-LAND TO CAL-MEN
    ELSE
    COMPUTE CAL-MEN = (MEN-PER-LAND + 1).
    MOVE CAL-RECORD TO GUA-LAND-RECORD.
    MOVE CAL-MEN TO MEN-PER-LAND.
    MOVE DATE-CARD TO UPDATE-DATE-D.
    MOVE SPACE TO BLANK-D.
REWRITE-F72.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC FROM HEAD-AFTER AFTER POSITIONING 1.
    REWRITE LAND-STAT-REC
        INVALID KEY GO TO INVA-REWRITEDA.
    IF CHKD NOT = '00' GO TO CHKRTND.
    MOVE SPACE TO L2.
    MOVE LAND-STAT-REC TO L2.
***** CHANGE APOF TO ITOF F7****
    MOVE '1' TO A-I-DW1.
    MOVE '100' TO CHECK-DW1.
    MOVE GUA-LAND-KEY-DW1 TO GUA-LAND-KEY-D.
    MOVE DATE-CARD TO UPDATE-DATE-DW1.
    READ LAND-STAT-FILE INVALID KEY GO TO CHECK-D23.
    IF CHKD NOT = '00' GO TO CHKRTNS.
    REWRITE LAND-STAT-REC FROM LAND-STAT-REC-W1
        INVALID KEY GO TO INVA-REWRITEDI.
    IF CHKD NOT = '00' GO TO CHKRTND.
    MOVE LAND-STAT-REC-W1 TO DETAIL-L.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
PIM-LAND.
    MOVE L2 TO DETAIL-L.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
**************************************************************
****CREATE ASSET FILE****
**************************************************************
CREATE-ASSET.
    ADD 1 TO CX.
    IF CX NOT   1 GO TO PREPARE-F6.
```

```
    IF SW-D = 1 MOVE DUM-LAND TO ID-GUA-LAND-CO-W1
                MOVE 1 TO SEQUENCE-A2W1
                PERFORM WRITE-F6 THRU WRITE-F6-EXIT.
    MOVE ID-GUA-TR TO ID-GUA-LAND-CO-W1.
    MOVE CAL-RECORD TO SEQUENCE-A2W1.
    PERFORM WRITE-F6 THRU WRITE-F6-EXIT.
PREPARE-F6.
    IF SW-D = 1  MOVE DUM-LAND TO ID-GUA-LAND-AW1
                 MOVE 1 TO SEQUENCE-A1W1 C-SEQUENCE
    ELSE MOVE ID-GUA-TR TO ID-GUA-LAND-AW1
    MOVE CAL-RECORD TO SEQUENCE-A1W1 C-SEQUENCE.
    MOVE '0' TO A-I-AW1.
    IF CX = 1 MOVE GUA-LAND-KEY-AW1 TO KEEPRECL.
    MOVE '110' TO CHECK-AW1.
    MOVE L-TYPE-TR TO L-TYPE-AW1.
    MOVE IDTR TO ID-MEMBER-AW1.
    MOVE PRINCIPAL-TR TO PRINCIPAL-AW1.
    MOVE GUA-LIMIT-TR TO GUA-LIMIT-AW1.
    MOVE ZEROES TO GUA-LAND-CO-KEY-W1.
    MOVE DATE-CARD TO UPDATE-DATE-AW1.
    MOVE SPACE TO BLANK-AW1.
    IF GUA-NUM-TR = SEQUENCE-NO GO TO WRITE-F6.
    GO TO READ-TRAN.
WRITE-F6.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC FROM HEAD-ASSET AFTER POSITIONING 2.
    MOVE 0 TO SW-A.
    MOVE ASSET-GUA-REC-W1 TO ASSET-GUA-REC.
    READ ASSET-GUA-FILE.
    IF CHKA = '23' PERFORM WA23 THRU WA23-EXIT.
    IF SW-A = 0 PERFORM RE23 THRU RE23-EXIT.
    MOVE ASSET-GUA-REC TO DETAIL-S.
    PERFORM N-PAGE THRU N-PAGE-EXIT
    WRITE PRINT-REC FROM DATA-64 AFTER POSITIONING 1.
    IF SW-D = 1 PERFORM D-DUMMY THRU D-DUMMY-EXIT.
WRITE-F6-EXIT.  EXIT.
PREPARETEMP.
    PERFORM READ-TEMPFILE THRU READ-TEMPFILE-EXIT.
    MOVE GUA-NUM-TR TO GUA-LAND-NUMB-T.
UPDATE-LOANFILE3.
    IF ORDINARY-L = '2'
        MOVE ORDINARY-L TO EXTRA-L
        GO TO CHANGE-3I.
    IF EXTRA-L = '0' OR EXTRA-L = L-TYPE-TR
        GO TO UPDATE-LOANFILE.
    GO TO CHANGE-3I.
DUMMY-LAND.
    MOVE 900000000010 TO GUA-LAND-KEY-A.
    READ ASSET-GUA-FILE INVALID KEY GO TO INVA-READAA.
    IF CHKA NOT = '00' GO TO CHKRTNA.
    MOVE ID-GUA-LAND-CO TO DUM-LAND.
    COMPUTE ID-GUA-LAND-CO = ID-GUA-LAND-CO - 1.
    MOVE DATE-CARD TO UPDATE-DATE-A.
    REWRITE ASSET-GUA-REC INVALID KEY GO TO INVA-REWRITEAA.
    IF CHKA NOT = '00' GO TO CHKRTNA.
    ADD 1 TO SW-D.
    MOVE DUM-LAND TO ID-GUA-LAND-A.
    MOVE    1       TO SEQUENCE-A1
    MOVE    '0'     TO A-I-A1.
    READ ASSET-GUA-FILE INVALID KEY GO TO CHECK-A23.
    IF CHKA NOT = '00' GO TO CHKRTNA.
    GO TO CREATE-ASSET.
D-DUMMY.
    WRITE PRINT-REC FROM DATA-64 AFTER POSITIONING 1.
    MOVE SPACE TO PRINT-REC.
    WRITE PRINT-REC AFTER POSITIONING 2.
    DISPLAY '**** WAIT FOR LOAN BECAUSE OF DUMMY LAND, ID-MEMBER*
-   '***' ID-TRAN.
D-DUMMY-EXIT. EXIT.
WA23.
    WRITE ASSET-GUA-REC FROM ASSET-GUA-REC-W1
        INVALID KEY GO TO INVA-WRITEAA.
    IF CHKA NOT = '00' GO TO CHKRTNA.
    ADD 1 TO SW-A.
WA23-EXIT. EXIT.
```

```
RE23.
    IF CHKA NOT = '00' GO TO CHKRTNA.
    REWRITE ASSET-GUA-REC FROM ASSET-GUA-REC-WI
        INVALID KEY GO TO INVA-REWRITEAA.
    IF CHKA NOT = '00' GO TO CHKRTNA.
RE23-EXIT. EXIT.
INVA-READSA.
    IF CHKS = '23'  PERFORM N-PAGE
        DISPLAY '****READ STAT-FILE NOT FOUND ID-MEMBER (A)'
-       ID-TRAN.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****READ STAT-FILE ERROR. STATUS KEY IS ' CHKS.
    GO TO P-STOP.
CHKRTNS.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKS.
    GO TO P-STOP.
INVA-READSI.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****READ STAT-FILE ERROR. STATUS KEY IS ' CHKS.
    GO TO P-STOP.
INVA-REWRITESA.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****REWRITE STAT-FILE (A) ERROR. STATUS KEY VALUE IS
-   ' ' CHKS.
    GO TO P-STOP.
INVA-REWRITESI.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****REWRITE STAT-FILE (I) ERROR. STATUS KEY VALUE IS
-   ' ' CHKS.
    GO TO P-STOP.
CHECK-S23.
    IF CHKS = '23' GO TO P-WRITES23.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****READ STAT-FILE ERROR. STATUS KEY IS ' CHKS.
    GO TO P-STOP.
P-WRITES23.
    WRITE STAT-REC FROM STAT-REC-WI
        INVALID KEY GO TO INVA-WRITESI.
    IF CHKS NOT = '00' GO TO CHKRTNS.
    MOVE STAT-REC-WI TO DETAIL-L
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
    GO TO PIN-ATNF.
INVA-READAA.
    IF CHKA = '23'
        DISPLAY '****READ ASSET-GUA-FILE NOT FOUND LAND-ID'
-   ID-GUA-TR.
    DISPLAY '****READ ASSET-GUA-FILE ERROR.STATUS KEY VALUE IS'
    CHKA.
    GO TO P-STOP.
INVA-READTA.
    DISPLAY '****READ TEMP-LOAN-FILE ERROR. STATUS KEY VALUE IS'
-    CHKT.
    GO TO P-STOP.
INVA-READLA.
    IF CHKL = '23' PERFORM N-PAGE
        DISPLAY '****READ LOAN-FILE NOT FOUND ID-MEMBER (A) '
        ID-MEMBER-L.
    DISPLAY '****READ LOAN-FILE ERROR. STATUS KEY VALUE ' CHKL.
    GO TO P-STOP.
INVA-WRITELA.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****WRITE LOAN-FILE (A) ERROR. STATUS KEY VALUE IS'
-   CHKL.
    GO TO P-STOP.
CHECK-L23.
    IF CHKL = '23' GO TO P-WRITEL23.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    DISPLAY '****READ LOAN-FILE ERROR. STATUS KEY VALUE ' CHKL.
    GO TO P-STOP.
P-WRITEL23.
    WRITE LOAN-REC FROM LOAN-REC-WI
        INVALID KEY GO TO INVA-WRITELI.
    IF CHKL NOT = '00' GO TO CHKRTNL.
    MOVE LOAN-REC-WI TO DETAIL-L.
    PERFORM N-PAGE THRU N-PAGE-EXIT.
    WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
    GO TO PIN-LOAN.
```

```
 INVA-WRITESI.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE STAT-FILE (I) ERROR. STATUS KEY VALUE IS'
-      CHKS.
      GO TO P-STOP.
 INVA-WRITELI.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE LOAN-FILE (A) ERROR. STATUS KEY VALUE IS'
-      CHKL.
      GO TO P-STOP.
 INVA-REWRITELI.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE LOAN-FILE (I) ERROR. STATUS KEY VALUE IS
-     ' ' CHKL.
      GO TO P-STOP.
 INVA-READPA.
      IF CHKP = '23'  PERFORM N-PAGE THRU N-PAGE-EXIT
          DISPLAY '****READ PERS-GUA-FILE NOT FOUND GUA-ID'
          ID-GUA-MEN-P.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ PERS-GUA-FILE ERROR. STATUS KEY VALUE IS'
-      CHKP.
      GO TO P-STOP.
 INVA-REWRITEPA.
      IF CHKP NOT = '21' GO TO CHKRTNPK.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE PERS-GUA-FILE (A) ERROR. STATUS KEY VALU
-     'E IS ' CHKP.
      GO TO P-STOP.
 CHECK-P23.
      IF CHKP = '23' GO TO P-WRITEP23.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ PERS-GUA-FILE ERROR. STATUS KEY VALUE IS'
-      CHKP.
      GO TO P-STOP.
 P-WRITEP23.
      MOVE '0' TO SW-1.
      GO TO P-KEYF5.
 CHECK-L232.
      IF CHKL = '23' GO TO P-WRITEL232.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ LOAN-FILE ERROR. STATUS KEY VALUE ' CHKL.
      GO TO P-STOP.
 P-WRITEL232.
      WRITE LOAN-REC FROM LOAN-REC-WI
          INVALID KEY GO TO INVA-WRITELI.
      IF CHKL NOT = '00' GO TO CHKRTNL.
      MOVE LOAN-REC-WI TO DETAIL-L.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
      GO TO CHANGE-TOAPOL4.
 INVA-READDA.
      IF CHKD = '23' PERFORM PREPARE-F7N THRU PREPARE-F7N-EXIT
          GO TO P-WRITEDA23.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ LAND-STAT FILE ERROR. STATUS KEY VALUE IS'
-      CHKD.
      GO TO P-STOP.
 P-WRITEDA23.
      WRITE LAND-STAT-REC INVALID KEY GO TO INVA-WRITEDA.
      IF CHKD NOT = '00' GO TO CHKRTND.
      MOVE LAND-STAT-REC TO DETAIL-L
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
      GO TO CREATE-ASSET.
 INVA-WRITEDA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE LAND-STAT-FILE(A) ERROR. STATUS KEY VALUE
-     'IS ' CHKD.
      GO TO P-STOP.
 INVA-REWRITEDA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE LAND-STAT FILE (A) ERROR. STATUS KEY VAL
-     'UE IS ' CHKD.
      GO TO P-STOP.
```

```
 CHECK-D23.
      IF CHKD = '23'  GO TO P-WRITEDI23.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ LAND-STAT-FILE ERROR. STATUS KEY VALUE IS'
-     CHKD.
      GO TO P-STOP.
 P-WRITEDI23.
      WRITE LAND-STAT-REC FROM LAND-STAT-REC-WI
          INVALID KEY GO TO INVA-WRITEDI.
      IF CHKD NOT = '00' GO TO CHKRTND.
      MOVE LAND-STAT-REC-WI TO DETAIL-L.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      WRITE PRINT-REC FROM DATA-128 AFTER POSITIONING 1.
      GO TO PIM-LAND.
 INVA-WRITEDI.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE LAND-STAT(I) ERROR. STATUS KEY VALUE IS'
-      CHKD.
      GO TO P-STOP.
 INVA-REWRITEDI.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE LAND-STAT (I) ERROR. STATUS KEY VALUE IS
-     ' ' CHKD.
      GO TO P-STOP.
 INVA-WRITEAA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE ASSET-GUA-FILE ERROR. STATUS KEY VALUE IS'
-      CHKA.
      GO TO P-STOP.
 INVA-REWRITEAA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE ASSET-GUA-FILE ERROR. STATUS KEY VALUE I
-     'S ' CHKA.
      GO TO P-STOP.
 CHECK-A23.
      IF CHKA = '23' GO TO CREATE-ASSET.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****READ ASSET-GUA-FILE ERROR. STATUS KEY VALUE IS'
-      CHKP.
      GO TO P-STOP.
 INVA-WRITEPA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE PERS-GUA-FILE (A) ERROR. STATUS KEY VALUE
-     'IS ' CHKP.
      GO TO P-STOP.
 INVA-REWRITELA.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE LOAN-FILE (A) ERROR. STATUS KEY VALUE IS
-     ' ' CHKL.
      GO TO P-STOP.
 CHKRTNL.
      DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKL.
      GO TO P-STOP.
 CHKRTNP.
      DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKP.
      GO TO P-STOP.
 CHKRTNA.
      DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKA.
      GO TO P-STOP.
 CHKRTND.
      DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKD.
      GO TO P-STOP.
 CHKRTNT.
      DISPLAY '****ERROR. STATUS KEY VALUE IS ' CHKT.
      GO TO P-STOP.
 CHKRTNPR.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****REWRITE PERS-GUA-FILE ERROR. STATUS KEY VALUE IS
-     ' ' CHKP.
      GO TO P-STOP.
 CHKRTNPW.
      PERFORM N-PAGE THRU N-PAGE-EXIT.
      DISPLAY '****WRITE PERS-GUA-FILE ERROR. STATUS KEY VALUE IS
-     ' ' CHKP.
      GO TO P-STOP.
 IN-REP.
      DISPLAY '*** REWRITE PERS-GUA-FILE ERROR. STATUS KEY VALUE
-     'IS  ' CHKP.
      GO TO P-STOP.
```

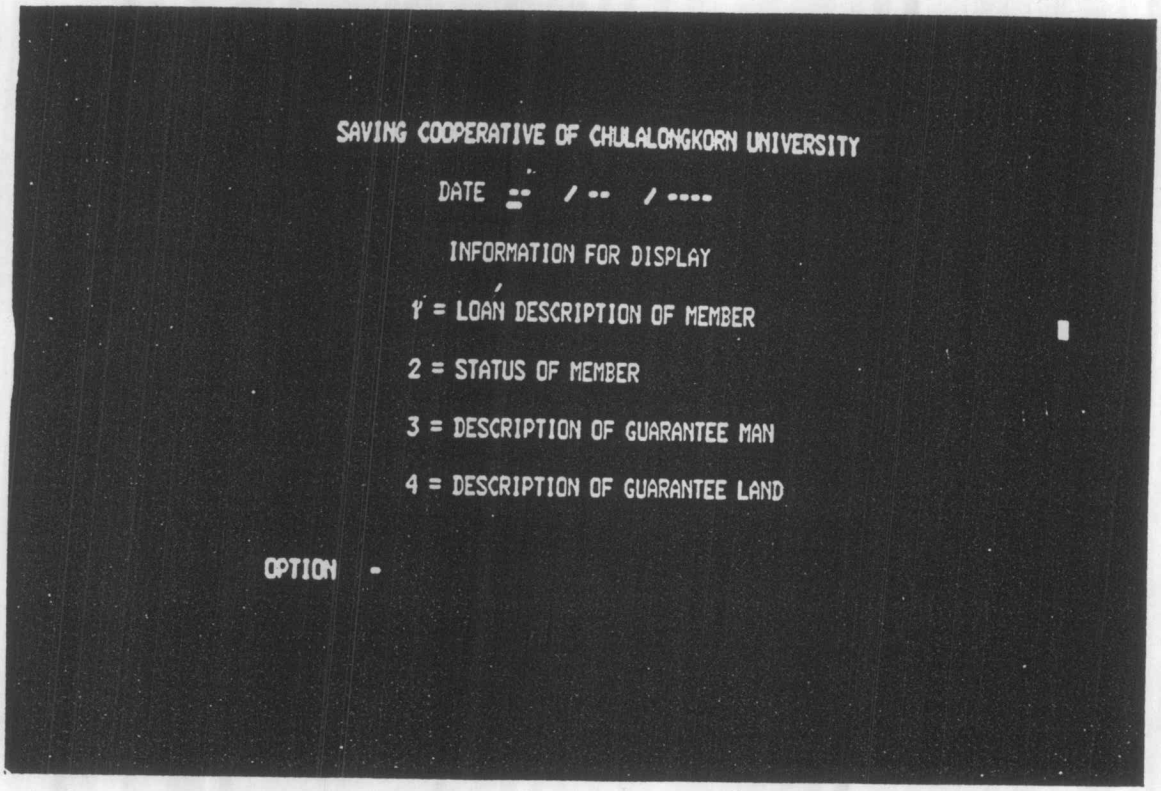
```
CHKRTN-S.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****OPEN STAT-FILE ERROR. STATUS KEY VALUE ' CHKS.
     GO TO P-STOP.
CHKRTN-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****OPEN LOAN-FILE ERROR. STATUS KEY VALUE ' CHKL.
     GO TO P-STOP.
CHKRTN-P.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****OPEN PERS-GUA-FILE ERROR. STATUS KEY VALUE IS'
-     CHKP.
     GO TO P-STOP.
CHKRTN-A.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '**** OPEN ASSET-GUA-FILE ERROR. STATUS KEY VALUE IS'
-     CHKA.
     GO TO P-STOP.
CHKRTN-D.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****OPEN LAND-STAT-FILE ERROR. STATUS KEY VALUE IS'
-    CHKD.
     GO TO P-STOP.
CHKRTN-T.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****OPEN TEMP-LOAN-FILE ERROR. STATUS KEY VALUE IS'
-    CHKT.
     GO TO P-STOP.
CHKRTNC.
     DISPLAY '****ERROR DATA-FILE,STATUS KEY VALUE IS ' CHKC.
     GO TO P-STOP.
CLOSE-S.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE STAT-FILE ERROR. STATUS KEY VALUE ' CHKS.
     GO TO P-FINIT.
CLOSE-L.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE LOAN-FILE ERROR. STATUS KEY VALUE IS'
-    CHKL.
     GO TO P-FINIT.
CLOSE-P.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE PERS-GUA-FILE ERROR. STATUS KEY VALUE '
-    CHKP.
     GO TO P-FINIT.
CLOSE-A.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE ASSET-GUA-FILE ERROR. STATUS KEY VALUE IS'
-     CHKA.
     GO TO P-FINIT.
CLOSE-D.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE LAND-STAT-FILE ERROR. STATUS KEY VALUE '
-    CHKD.
     GO TO P-FINIT.
CLOSE-T.
     PERFORM N-PAGE THRU N-PAGE-EXIT.
     DISPLAY '****CLOSE TEMP-LOAN-FILE ERROR. STATUS KEY VALUE IS'
-     CHKT.
     GO TO P-FINIT.
CLOSE-C.
     DISPLAY '****CLOSE DATA-FILE ERROR.STATUS KEY VALUE IS ' CHKC
     GO TO P-FINIT.
END-RTN.
     MOVE COUNT-TRAN TO NO-TRAN.
     MOVE SPACE TO PRINT-REC.
     MOVE ALL '-' TO L1.
     WRITE PRINT-REC FROM HEAD-LINE AFTER POSITIONING 2.
     MOVE DUM TO NO-DUM.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM HEAD-DUM AFTER POSITIONING 2.
     MOVE SPACE TO PRINT-REC.
     WRITE PRINT-REC FROM LAST-LINE AFTER POSITIONING 2.
```

```
P-STOP.
    CLOSE STAT-FILE.
    IF CHKS NOT = '00' GO TO CLOSE-S.
    CLOSE LOAN-FILE.
    IF CHKL NOT = '00' GO TO CLOSE-L.
    CLOSE PERS-GUA-FILE.
    IF CHKP NOT = '00' GO TO CLOSE-P.
    CLOSE ASSET-GUA-FILE.
    IF CHKA NOT = '00' GO TO CLOSE-A.
    CLOSE LAND-STAT-FILE.
    IF CHKD NOT = '00' GO TO CLOSE-D.
    CLOSE TEMP-LOAN-FILE.
    IF CHKT NOT = '00' GO TO CLOSE-T.
    CLOSE DATA-FILE.
    IF CHKC NOT = '00' GO TO CLOSE-C.
P-FINIT.
    STOP RUN.
```

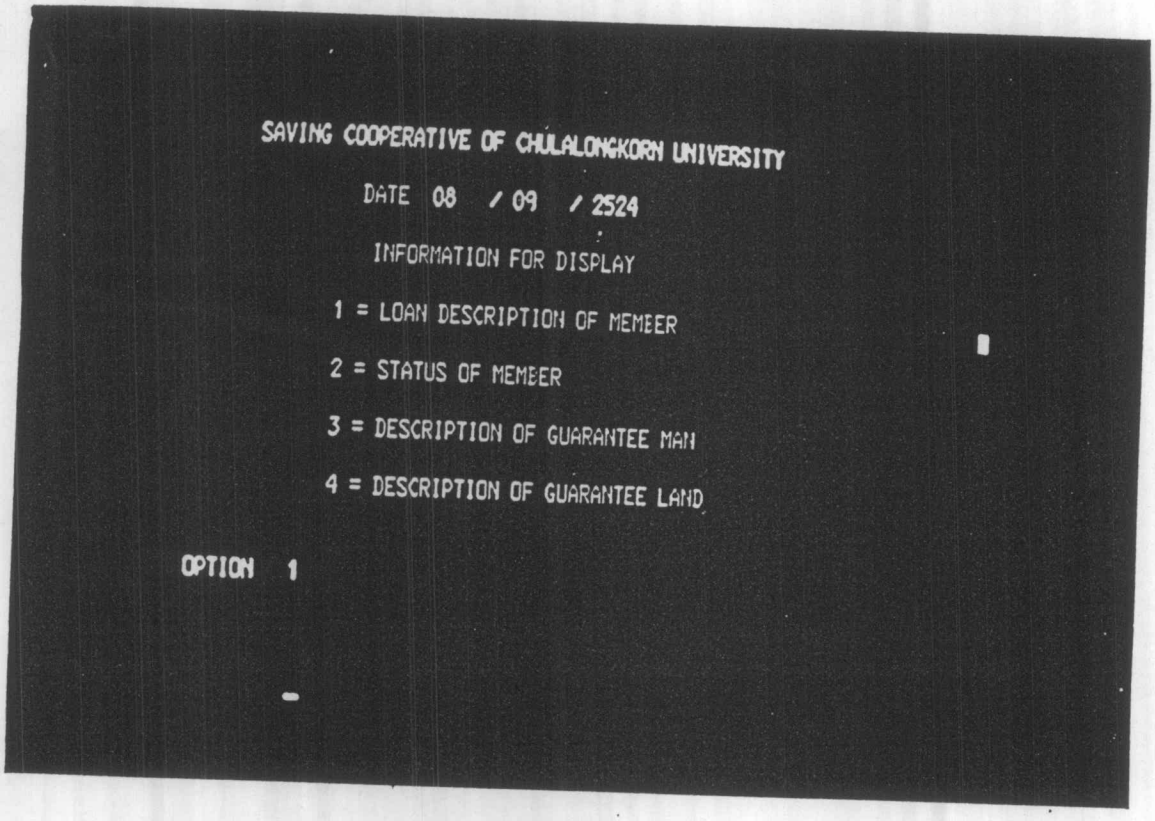
ภาคผนวก ง.

รายงานที่ได้จากโปรแกรมที่ใช้ในการวิจัย

ง.๑ จอภาพแสดงข้อมูลนำเข้าของจอภาพ RO1

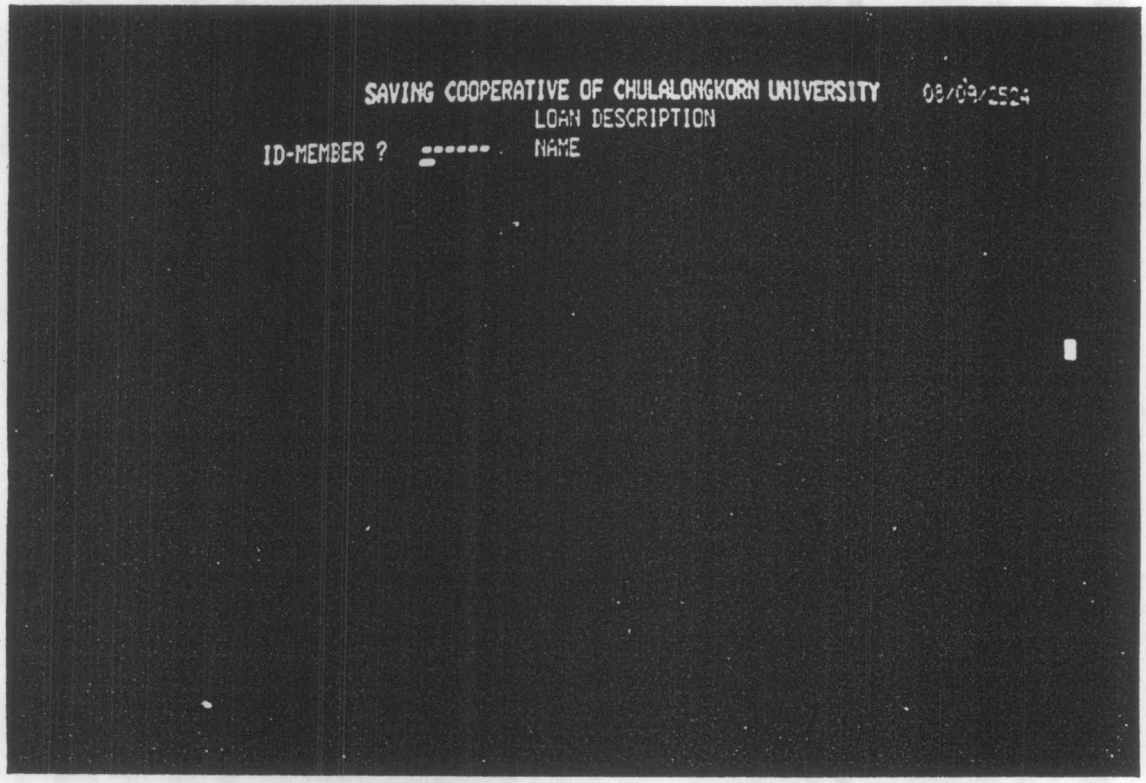


ง.๒ ตัวอย่างรายงานที่แสดงบนจอภาพแสดงลักษณะงานที่ต้องการ (RO1)

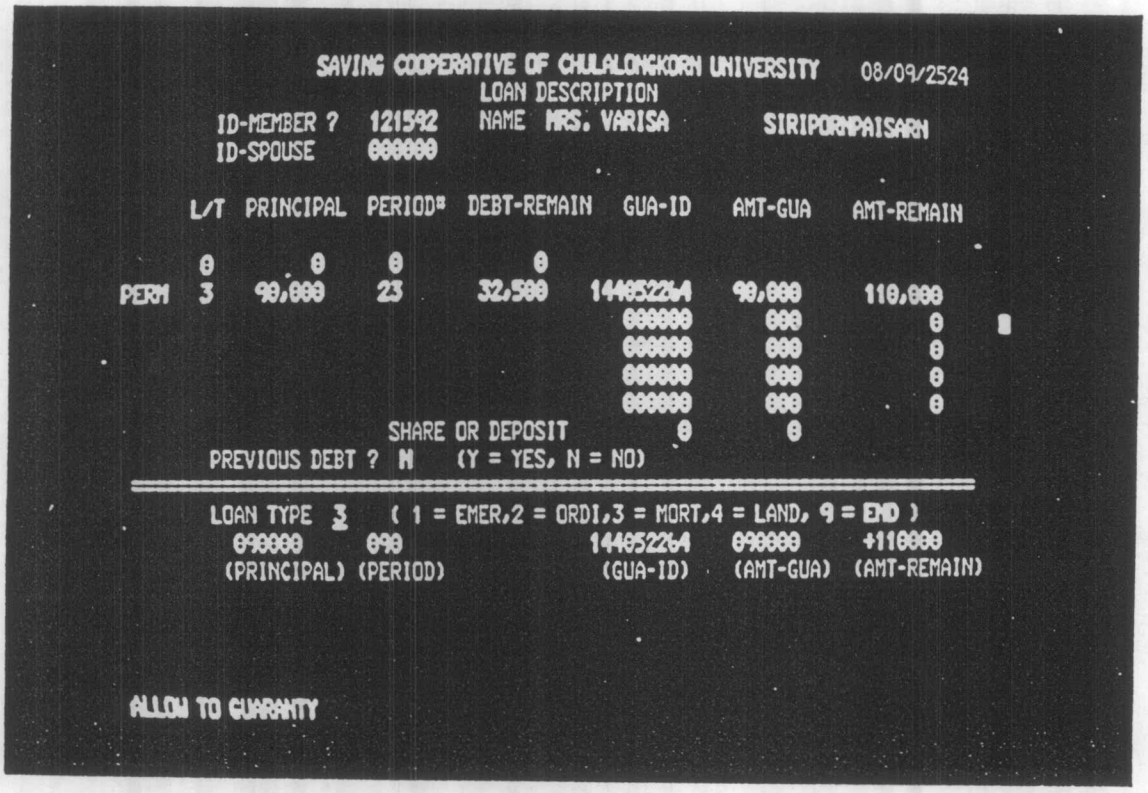


ง.๓ จอภาพแสดงข้อมูลนำเข้าของจอภาพ R02

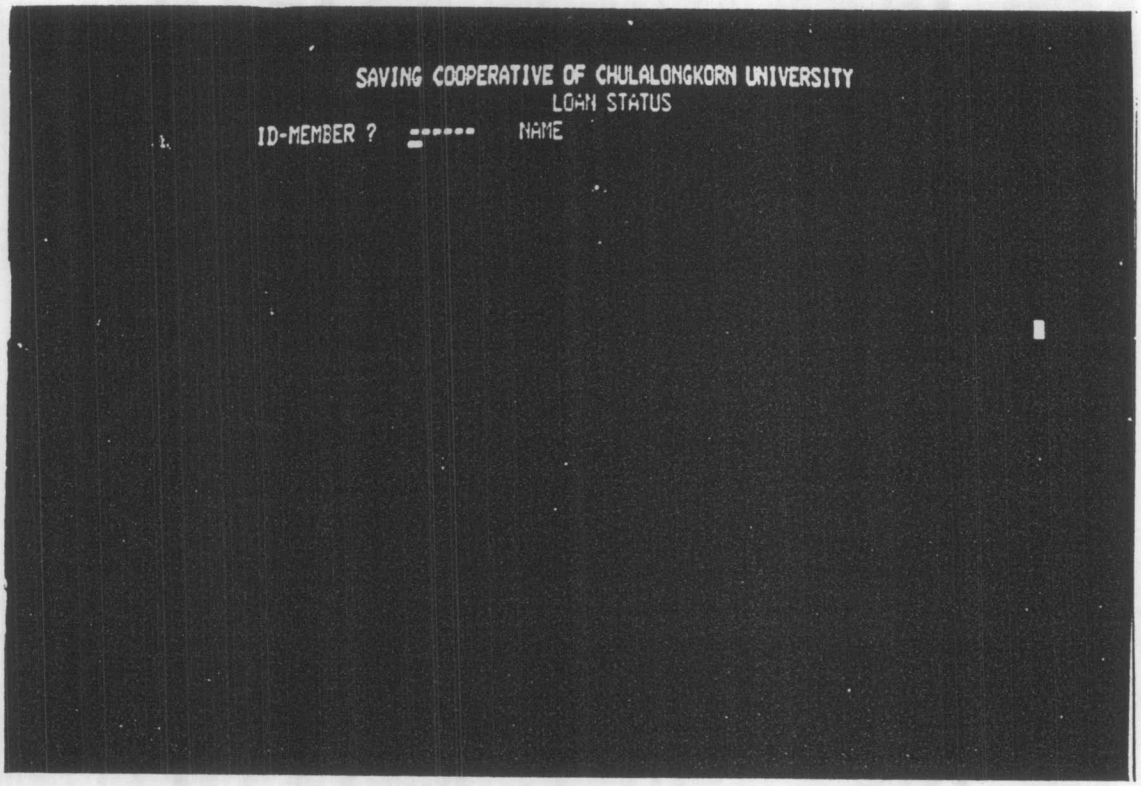
หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย



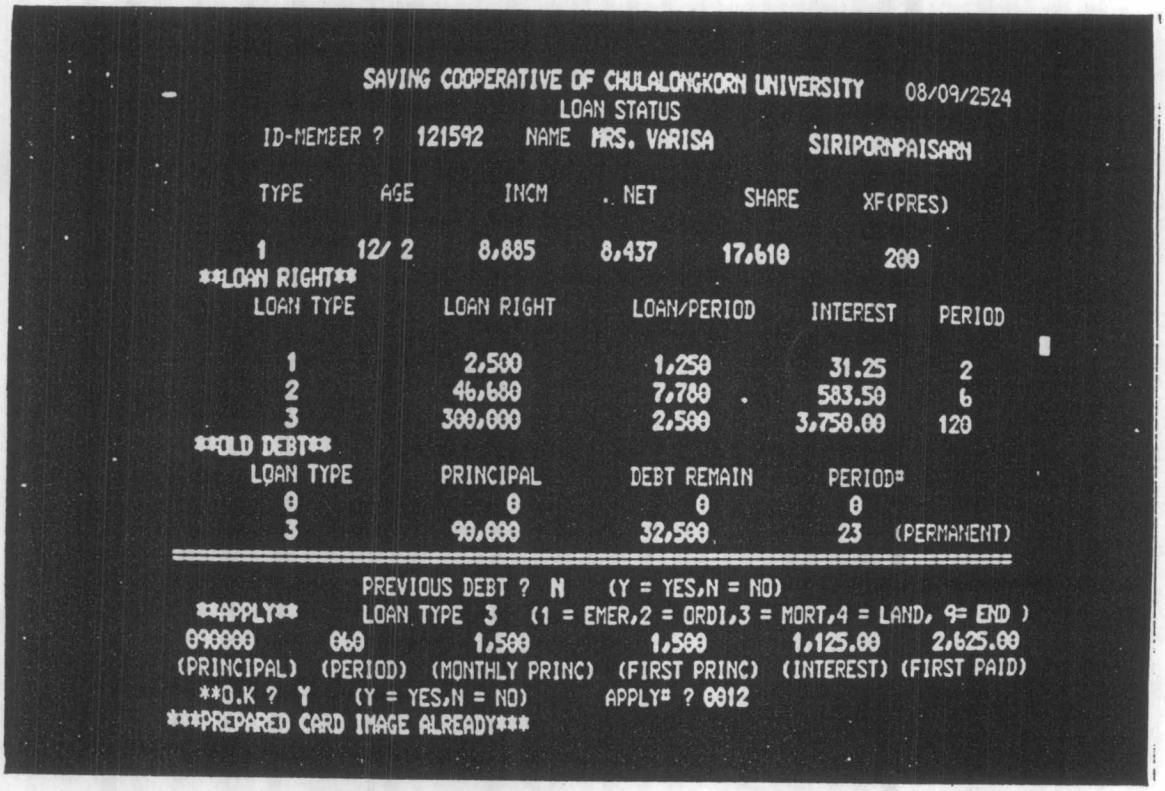
ง.๔ ตัวอย่างรายงานที่แสดงบนจอภาพแสดงรายละเอียดการกู้ของสมาชิก (RO2)



ง.๕ จอภาพแสดงข้อมูลนำเข้าของจอภาพ R03

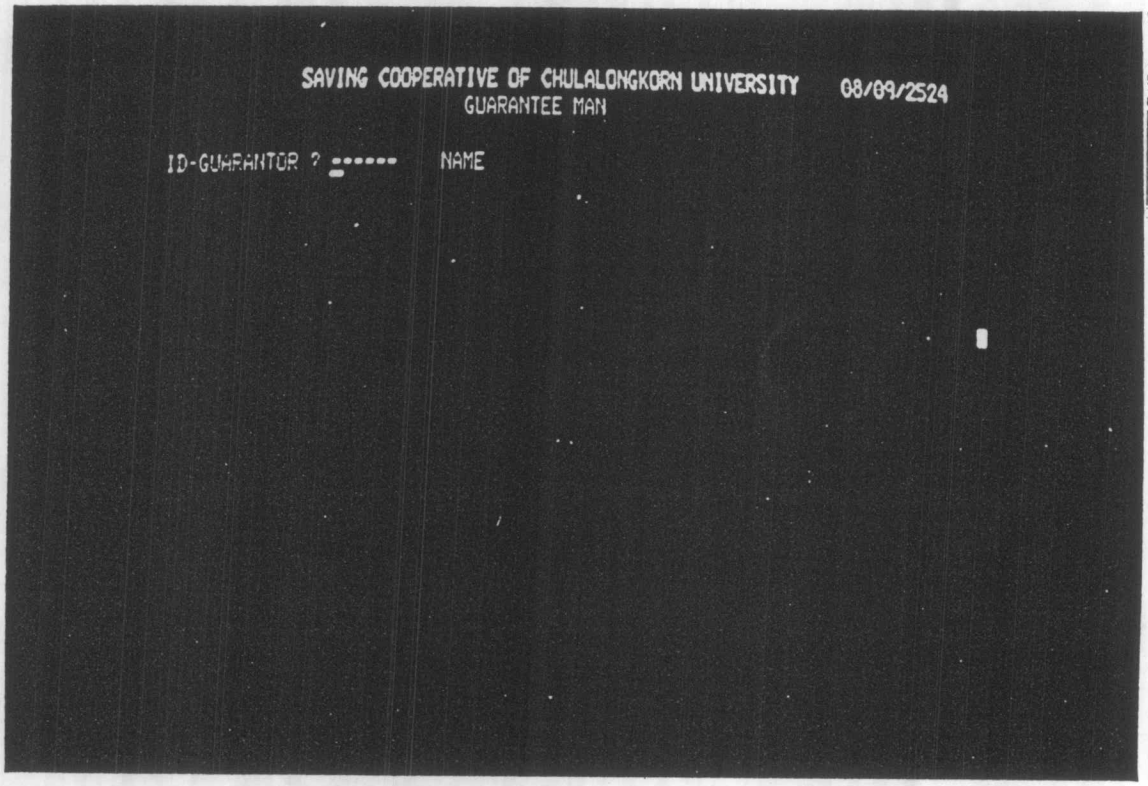


ง.๖ ตัวอย่างรายงานที่แสดงบนจอภาพแสดงสถานภาพของสมาชิก (R03)

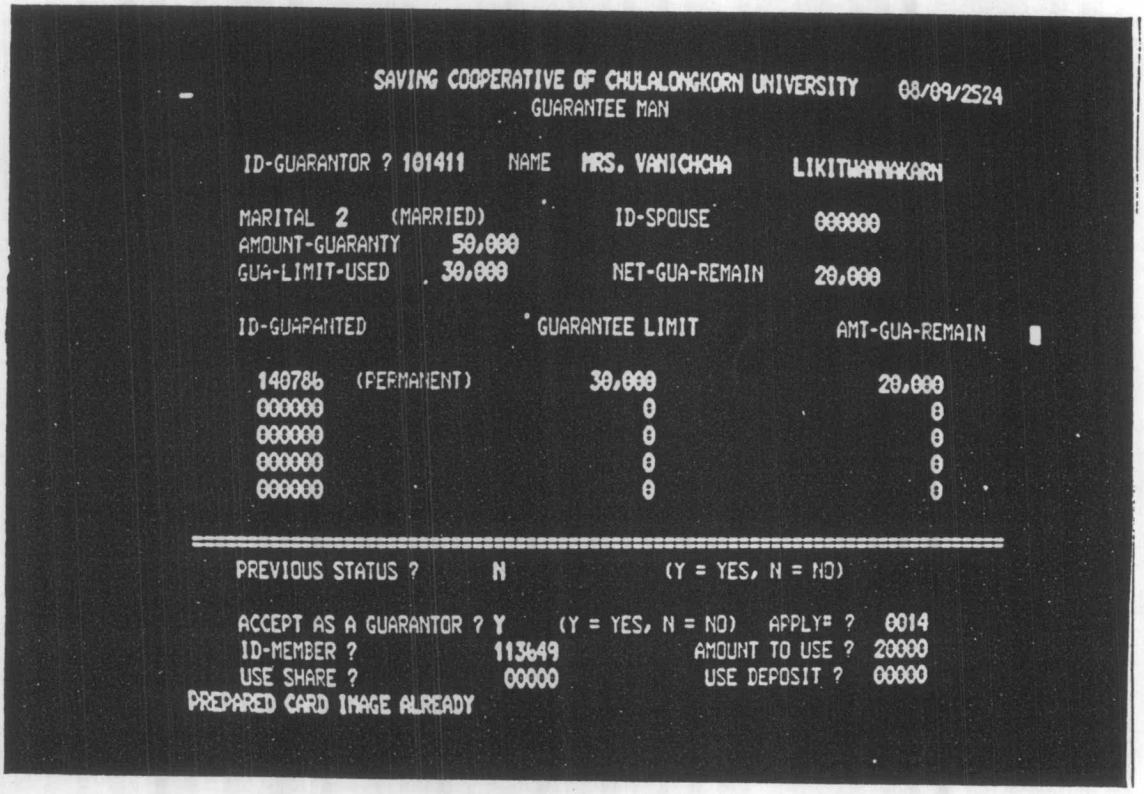


```
                SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY   08/09/2524
                                LOAN STATUS
      ID-MEMBER ?  121592    NAME  MRS. VARISA          SIRIPORNPAISARN

      TYPE      AGE       INCM      NET       SHARE      XF(PRES)

        1       12/ 2     8,885     8,437     17,610       200
    **LOAN RIGHT**
       LOAN TYPE        LOAN RIGHT       LOAN/PERIOD      INTEREST    PERIOD

           1              2,500             1,250           31.25        2
           2             46,680             7,780           583.50       6
           3            300,000             2,500         3,750.00     120
    **OLD DEBT**
       LOAN TYPE        PRINCIPAL        DEBT REMAIN       PERIOD*
           0                0                 0              0
           3             90,000            32,500            23     (PERMANENT)
═══════════════════════════════════════════════════════════════════════════
                PREVIOUS DEBT ?  N     (Y = YES,N = NO)
     **APPLY**      LOAN TYPE  3   (1 = EMER,2 = ORDI,3 = MORT,4 = LAND, 9= END )
     090000       060          1,500          1,500        1,125.00     2,625.00
    (PRINCIPAL)  (PERIOD)  (MONTHLY PRINC)  (FIRST PRINC)  (INTEREST) (FIRST PAID)
      **O.K ?  Y     (Y = YES,N = NO)        APPLY# ? 0012
    ***PREPARED CARD IMAGE ALREADY***
```

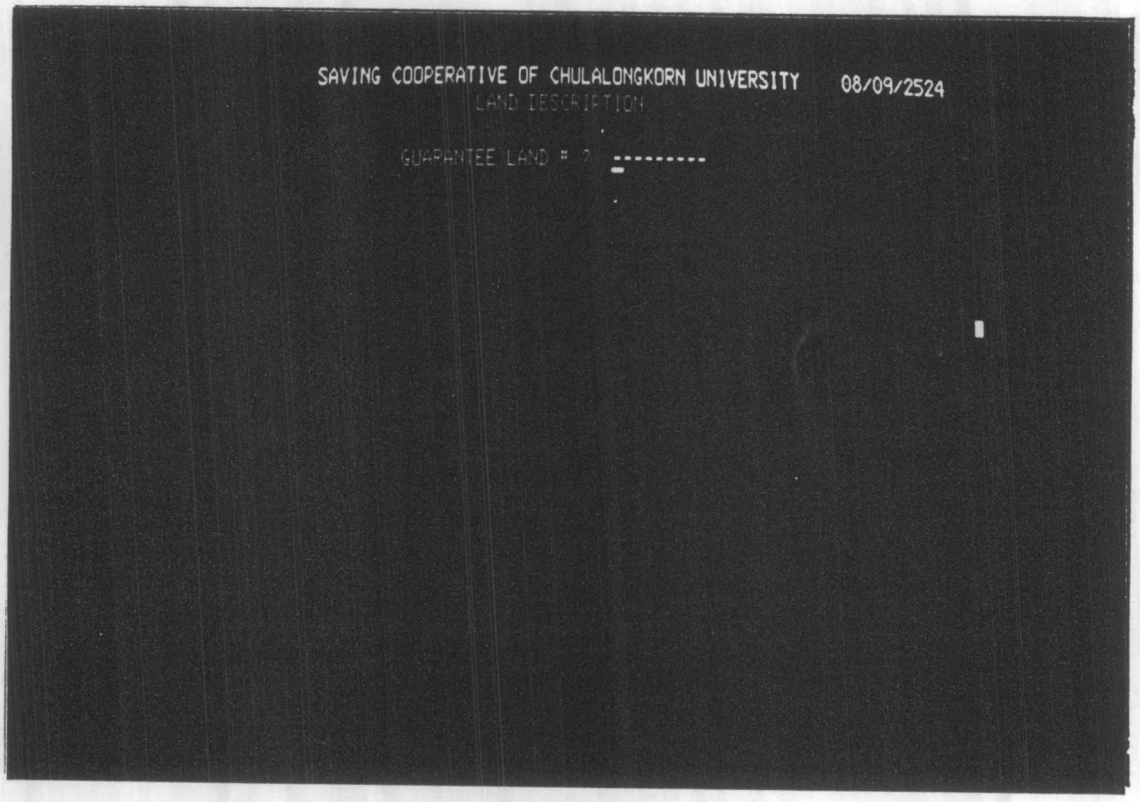
ง.๗ จอภาพแสดงข้อมูลนำเข้าของจอภาพ R04



ง.๘ ตัวอย่างรายงานที่แสดงบนจอภาพแสดงความสามารถในการค้ำประกันของสมาชิก (RO4)



ง.๙ จอภาพแสดงข้อมูลนำเข้าของจอภาพ RO5



ง.๑๐ ตัวอย่างรายงานที่แสดงบนจอภาพแสดงรายละเอียดของหลักทรัพย์ค้ำประกัน

(R05)

```
                SAVING COOPERATIVE OF CHULALONGKORN UNIVERSITY   00/00/2524
                          LAND DESCRIPTION

                      GUARANTEE LAND # ? 14405226A
  LOAN TYPE 3  = EXTRA (MORTGAGE)      MEN PER LAND    1

  LAND-AREA   EVAL/UNIT   EVAL-PRICE   EVAL-BUILDING   FIRE INSURANCE
     50         2,000       100,000       300,000         300,000

  EVALUATE-DATE                          09/10/2522
  AMOUNT GUARANTEE SECURITIES              200,000
  REMAIN SECURITIES                        110,000
ID-GUARANTED              PRINCIPAL     SECUR-USED      OTHER SECUR
 121592   (PERMANENT)      90,000         90,000             0
 000000                         0              0             0
 000000                         0              0             0
 000000                         0              0             0
 000000                         0              0             0

          PREVIOUS STATUS ? N   (Y = YES,N = NO)
ACCEPT THIS LAND TO GUARANTEE ? Y   (Y = YES,N = NO)  APPLY# ?  0012
                   ID-MEMBER ? 121592          AMOUNT TO USE ? 000000
                   USE SHARE ? 000000           USE DEPOSIT ? 000000
***PREPARED CARD IMAGE ALREADY***
```

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

PREPARE AND UPDATE LOAN STATUS

DATE 10/09/2524 PAGE 7

****TRANSACTION****

252409000712245102030000030000211237120100000000000000000000000000000000000000000000000025240910

ID-MEMBER STATUS FILE

BEFORE
2245102000025221130023100000030030010046650020455500000000044650200015000187000000000000000025240810
AFTER
2245101100252211300231000000300300100466500204555000000000446502000150001870000000000000000252409100
2245101101252211300231000000300300110466500204555000000000446502000150001870000000000000000252409100

GUARANTEE-PERSON STATUS FILE

BEFORE
2112370000252100331043200000045008020413320020133200000000013950270005000078020000000000000025240810
AFTER
2112371100252100331043200000045008020413320020133200000000013950270005000078020000000000000025240910
2112370110252100331043200000045008020413320020133200100001013950270005000078020000000000000025240910

****TRANSACTION****

252409000722245102030000030000211255120200000000000000000000000000000000000000000000000025240910

GUARANTEE-PERSON STATUS FILE

BEFORE
2112550000252100331043218171015015025219760004197500000000021450200010000000000000000000000025240810
****FIND NEW GUARANTOR BECAUSE AMOUNT-GUARANTEE IS NOT ENOUGH000211255

GUARANTEE-PERSON FILE
211237101102245101000999997125240910

GUARANTEE-PERSON FILE
999997101102245102000000000025240910

TEMPORARY LOAN FILE
22451020110 252408080300001010000100002900003000103750000 01250150 000000000000000000000000000025240910

LOAN FILE
BEFORE
2245102000001550232523110103000010100001000019000030011037500025000012501500270000130280100000000000000000000025240810
AFTER
2245102110001550232523110103000010100001000019000030011037500025000012501500270000130280100000000000000000000025240910
2245102011000 252408080300001010000100002900003000103750000 01250150 211237100000000000000000000000025240910

NUMBER OF DUMMY GUARANTOR = 2

NUMBER OF UPDATED TRANSACTION = 7

ตัวอย่างรายงานการปรับปรุงสถานะเงินกู้ใหม่ (ROG)

ภาคผนวก จ.

คำศัพท์ที่ใช้ในการวิจัย

ศัพท์

ไทย	อังกฤษ	หน้า
การสื่อสารข้อมูล	Data Communication	๓๗
กำหนดคลัสเตอร์	Define Cluster	๓๘
ความปลอดภัย	security	๔๔
ความปลอดภัยของระบบ	System Security	๔๔
คอนโทรล	control	๓๙
คีย์	key	๔๔
แคตตาล็อค	catalog	๓๘
เคอร์เซอร์	cursor	๔๔
โคบอล	COBOL	๓๙
จอภาพ	Visual Display	๔๔
เจ้าหน้าที่ที่ใช้เทอร์มินัล	operator	๔๔
ซีพียู	CPU (Control Processing Unit)	๓๙
ซีไอซีเอส	CICS (Customer Information Control System)	๓๗
ดีเซ็ค	DSECT	๓๘
ฐานข้อมูล	Data Base	๓๗
เทอร์มินัล	terminal	๓๗
แบทช์	batch	๔๔
ไบท์	byte	๔๒
ผู้ใช้	user	๔๔
ฟังก์ชันคีย์	function key	๔๔
เมนู	menu	๔๔
แม็บ	MAP	๓๘
ยูทิลิตีโปรแกรม	Utility Program	๔๔

ไทย	อังกฤษ	หน้า
รหัสผ่าน	password	๔๔
ระบบการประมวลผลข้อมูลแบบสื่อสาร	On-line process system	๓๗
ระบบถามตอบออนไลน์	On-line inquiry system	๓๘
ระเบียนชั่วคราว	temporary record	๖๐
ระเบียนถาวร	permanent record	๖๐
วีแซม	VSAM(Virtual Storage Access Method)	๓๘
สลิป	slip	๑๕
อัพเดท	update	๖๘
ออนไลน์	on-line	๒
แอสเสสเมธอดเซอร์วิสโปรแกรม	Access Method Service Program	๕๕

ประวัติการศึกษา

นางสาวกัลยาณี กิตติศาสตรา เกิดวันที่ ๕ มีนาคม ๒๕๐๑ ที่จังหวัดพิษณุโลก สำเร็จการศึกษาวิทยาศาสตรบัณฑิต (สถิติ) จากคณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ พ.ศ. ๒๕๒๒ และเข้าศึกษาต่อในระดับปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. ๒๕๒๒