



บทที่ 6

ผลการวิจัย

ผลการวิจัยนี้ ได้ทดลองใช้โปรแกรมที่พัฒนาขึ้นกับ โปรแกรมประยุกต์ ไมโครซอฟต์ที่ โปรแกรมประยุกต์ ไมโครซอฟต์ฟอร์แทรน และ โปรแกรมประยุกต์ ไมโครซอฟต์ปาสคาล โดยแต่ละ โปรแกรมจะมีการทำงานเหมือนกัน และมีการเรียกใช้จอภาพที่ออกแบบร่วมกัน

ตัวอย่าง โปรแกรมประยุกต์ เป็น โปรแกรมสำหรับป้อนข้อมูลของระบบการลงทะเบียนเรียน ซึ่งประกอบด้วยจอภาพของการป้อนข้อมูลดังนี้

1. จอภาพการใส่รหัสผ่าน
2. จอภาพแสดงเมนูของระบบการลงทะเบียนเรียน เพื่อให้เลือกข้อมูลที่จะป้อน ได้แก่
 - 2.1 ประวัตินักศึกษา
 - 2.2 รายวิชา
3. จอภาพการป้อนข้อมูลประวัตินักศึกษา ได้แก่ ประวัติส่วนตัวและประวัติการศึกษา
4. จอภาพการป้อนข้อมูลรายวิชา

6.1 ตัวอย่างการออกแบบจอภาพ

จากการกำหนดจอภาพข้างต้น จะสร้างแฟ้มข้อมูลของจอภาพ โดยใช้โปรแกรมออกแบบจอภาพได้ดังนี้

6.1.1 แฟ้มข้อมูลของจอภาพการใส่รหัสผ่านชื่อ PASSWORD.SCN การกำหนดจอภาพนี้จะเน้นเรื่อง การกำหนดคุณสมบัติของเขตข้อมูลนำเข้าเป็นแบบไม่แสดงข้อมูลที่กำลังป้อน และการแสดงข่าวสารบนจอภาพ

6.1.2 แฟ้มข้อมูลของจอภาพแสดงเมนูการลงทะเบียนเรียนชื่อ STDMAIN.SCN การกำหนดจอภาพนี้จะเน้นเรื่อง การกำหนดเขตข้อมูลให้เป็นเมนูซึ่งสามารถเลือกได้

6.1.3 แฟ้มข้อมูลของจอภาพการป้อนข้อมูลประวัติส่วนตัวชื่อ STDRSUME.SCN การกำหนดจอภาพนี้จะเน้นเรื่อง การกำหนดคุณสมบัติต่าง ๆ ของเขตข้อมูลนำเข้า การตรวจสอบข้อมูลภายในช่วงที่กำหนด การตรวจสอบข้อมูลที่เก็บไปได้ และการกำหนดให้แสดงจอภาพถัดไปต่อเนื่องกัน

6.1.4 แฟ้มข้อมูลของจอภาพการป้อนข้อมูลประวัติการศึกษาชื่อ STDEDU.SCN การกำหนดจอภาพนี้จะเน้นเรื่อง การกำหนดให้แสดงจอภาพก่อนหน้าต่อเนื่องกัน

6.1.5 แฟ้มข้อมูลของจอภาพการป้อนข้อมูลรายวิชาชื่อ COURSE.SCN การกำหนดจอภาพนี้จะเน้นเรื่อง การกำหนดเขตข้อมูลนำเข้าซึ่งเป็นตัวแปรแถวลำดับให้สามารถเลื่อนดูที่ละบรรทัดและที่ละหน้าได้

6.2 ตัวอย่างโปรแกรมประยุกต์

ตัวอย่างของโปรแกรมประยุกต์สำหรับการป้อนข้อมูลของระบบการลงทะเบียนเรียนเป็นโปรแกรมทำหน้าที่แสดงเมนูให้ผู้เลือกใช้กว่าต้องการป้อนข้อมูลประวัตินักศึกษาหรือรายละเอียดรายวิชา เมื่อผู้ใช้เลือกแล้วก็จะแสดงจอภาพสำหรับป้อนข้อมูลนั้น ภายในโปรแกรมแต่ละภาษาจะมีการเรียกใช้จอภาพที่ออกแบบด้วยการเขียนคำสั่ง เรียกใช้ฟังก์ชันจัดการบนจอภาพผ่านโปรแกรมเชื่อมโยงของโปรแกรมแต่ละภาษา ตัวอย่างโปรแกรมประยุกต์ประกอบด้วย

6.2.1 โปรแกรม REGISTC.C เป็นโปรแกรมไมโครซอฟต์ซี

6.2.2 โปรแกรม REGISTP.PAS เป็นโปรแกรมไมโครซอฟต์ปาสคาล

6.2.3 โปรแกรม REGISTF.FOR เป็นโปรแกรมไมโครซอฟต์ฟอร์แทรน

แต่ละโปรแกรมมีขั้นตอนการทำงานที่เหมือนกันดังนี้

1. กำหนดตัวแปรทุกตัวแปรที่มีใช้ในแฟ้มข้อมูลของจอภาพทุกแฟ้มข้อมูล เพื่อให้โปรแกรมแสดงจอภาพรู้จักตัวแปรเหล่านี้ โดยใช้ฟังก์ชัน VDEFINE และ VDEFARR
2. แสดงจอภาพการใส่รหัสผ่าน โดยใช้ฟังก์ชัน DISPLAY
3. เก็บรายละเอียดของแฟ้มข้อมูลของจอภาพชื่อ STDMAIN.SCN STDRSUME.SCN STDEDU.SCN และ COURSE.SCN ไว้ในหน่วยความจำตลอดการทำงานของโปรแกรม โดยใช้ฟังก์ชัน SBUFFER

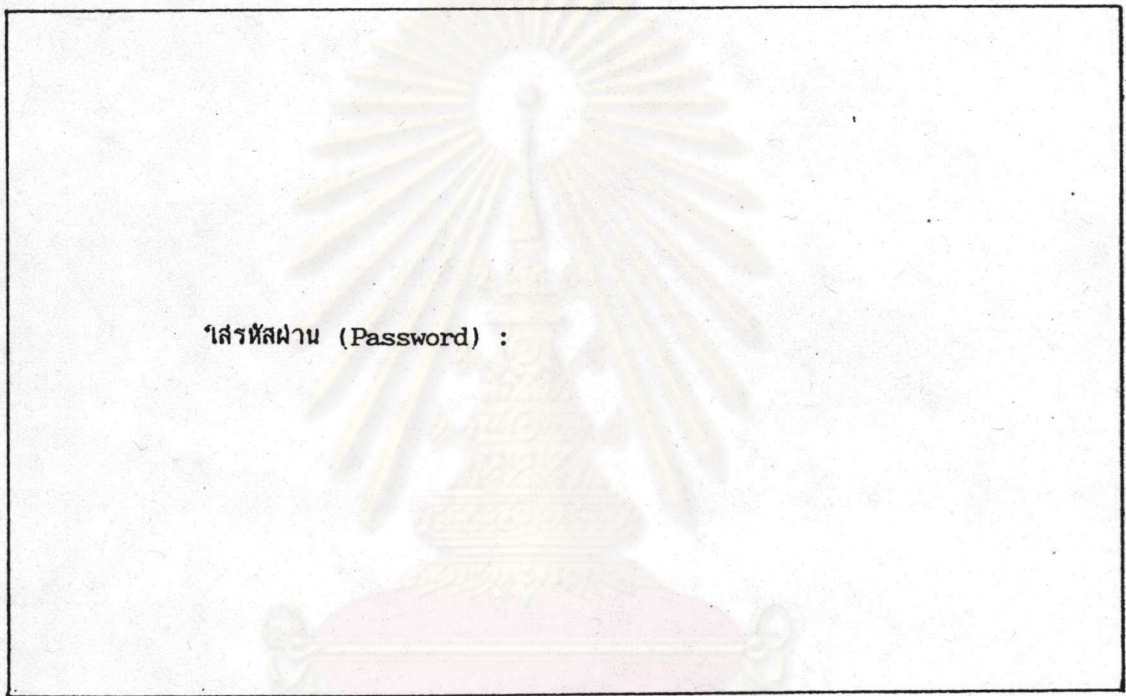
4. แสดงจอภาพเมนูหลักของระบบการลงทะเบียนชื่อ STDMAIN.SCN เพื่อให้ผู้ใช้เลือก ดังนี้

4.1 เลือกข้อ 1 โปรแกรมจะแสดงจอภาพการป้อนข้อมูลประวัตินักศึกษา ได้แก่จอภาพ STDRSUME.SCN และ STDEDU.SCN

4.2 เลือกข้อ 2 โปรแกรมจะแสดงจอภาพการป้อนข้อมูลรายวิชาชื่อ COURSE.SCN

วิธีการใช้คำสั่งเพื่อปฏิบัติการโปรแกรมประยุกต์ที่บรรทัดคำสั่ง ในที่นี้จะยกตัวอย่างโปรแกรม REGISTC.C ดังนี้

SCDP REGISTC.EXE /S3 /P2 /M3 /N5



ใส่รหัสผ่าน (Password) :

ศูนย์วิทยทรัพยากร
ภาพที่ 6.1 ตัวอย่างรายงานรายละเอียดของจอภาพ PASSWORD.SCN
จุฬาลงกรณ์มหาวิทยาลัย

1. Screen Definition

Screen Name : PASSWORD.SCN

Size of Screen File = 149 bytes

Start Location : 0, 0

End Location : 23,79

No of text field = 0

No of output field = 1

No of input field = 1

Cursor Field Name : -

Message Field Name : MESS1

Previous Screen Name : -

Next Screen Name : -

2. Field Definition

Name	Row,Col	Size	Field Attribute
PASSWD	8,41	1x 8	Reverse video Input Field Must Fill No Echo Help Message : ใส่รหัสผ่านยาว 8 ตัวอักษร
MESS1	14,25	1x35	Highlight Output Field

Left Justify None
Auto Skip Uppercase
Screen Exit

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ระบบการลงทะเบียนเรียน
การป้อนข้อมูล

1. ประวัติผลิต
 2. รายละเอียดรายวิชา
- Esc หยุดการทำงาน

เลือกหัวข้อที่ต้องการ แล้กด F2

ภาพที่ 6.3 ตัวอย่างรายงานรายละเอียดของจอภาพ STDMAIN.SCN

1. Screen Definition

Screen Name : STDMAIN.SCN

Size of Screen File = 286 bytes

Start Location : 0, 0

End Location : 23,79

No of text field = 4

No of output field = 0

No of input field = 0

Cursor Field Name : -

Message Field Name : -

Previous Screen Name : -

Next Screen Name : -

2. Field Definition

Name	Row,Col	Size	Field Attribute
HEAD	2,25	1x25	Reverse video Text Field
ENMNU1	8,30	1x12	Underline Text Field Selection Field State Field Name.....ENMENU First Character.....1
ENMNU2	10,30	1x17	Underline Text Field Selection Field State Field Name.....ENMENU First Character.....2
ENMNU3	12,30	1x12	Underline Text Field Selection Field State Field Name.....ENMENU First Character.....

ประวัติผลิต ประวัติส่วนตัว	
เลขประจำตัว	
ชื่อ/นามสกุล	
เพศ (F = หญิง , M = ชาย)	วัน/เดือน/ปีเกิด / /
ที่อยู่ปัจจุบัน	
กด F2 เพื่อบันทึกข้อมูลต่อ	Esc เพื่อยกเลิก

ศูนย์วิทยทรัพยากร
ภาพที่ 6.5 ตัวอย่างรายงานรายละเอียดของจอภาพ STDRSUME.SCN
จุฬาลงกรณ์มหาวิทยาลัย

1. Screen Definition

Screen Name : STDRSUME.SCN

Size of Screen File = 423 bytes

Start Location : 0, 0

End Location : 23,79

No of text field = 1

No of output field = 0

No of input field = 9

Cursor Field Name : -

Message Field Name : -

Previous Screen Name : -

Next Screen Name : STDEDU

2. Field Definition

Name	Row,Col	Size	Field Attribute		
HEAD	1,31	1x16	Reverse video Text Field		
STDID	5,16	1x 7	Reverse video Input Field Must Fill Uppercase	Left Justify Auto Skip	Alphanumeric Copy Value
STDNM	7,18	1x50	Reverse video Input Field Auto Skip	Left Justify Copy Value	None
SEX	9, 8	1x 1	Reverse video Input Field Auto Skip Possible Check F M	Left Justify Copy Value	Alphabetic Uppercase
BDAY	9,58	1x 2	Reverse video Input Field Must Fill Range Check Min Value = Max Value =	Left Justify Auto Skip	Integer Copy Value
BMON	9,61	1x 2	Reverse video Input Field Must Fill Range Check Min Value = Max Value =	Left Justify Auto Skip	Integer Copy Value
BYEAR	9,64	1x 2	Reverse video Input Field Must Fill Range Check Min Value = Max Value =	Left Justify Auto Skip	Integer Copy Value
STDAD1	11,22	1x50	Reverse video Input Field Auto Skip	Left Justify Copy Value	None
STDAD2	12,15	1x50	Reverse video Input Field Auto Skip	Left Justify Copy Value	None
STDAD3	13,15	1x50	Reverse video Input Field Auto Skip	Left Justify Copy Value	None

ประวัติการศึกษา		
วัน/เดือน/ปี ที่เข้าศึกษา	/ /	
สถานภาพการศึกษา	1 = นิสิตสามัญ	2 = นิสิตทดลองเรียน
จบการศึกษาจากโรงเรียน		
กค F2 - ถูกต้อง	F3 - กลับไปบอข้อมูลประวัติส่วนตัว	ESC - ยกเลิก

ภาพที่ 6.7 ตัวอย่างรายงานรายละเอียดของจอภาพ STDEDU.SCN

1. Screen Definition

Screen Name : STDEDU.SCN
 Size of Screen File = 372 bytes
 Start Location : 0, 0 End Location : 23,79
 No of text field = 0
 No of output field = 0
 No of input field = 5
 Cursor Field Name : - Message Field Name : -
 Previous Screen Name : STDRSUME Next Screen Name : -

2. Field Definition

Name	Row,Col	Size	Field Attribute		
EDAY	5,32	1x 2	Reverse video Input Field Auto Skip Range Check Min Value = Max Value =	Left Justify Copy Value	Integer
EMON	5,35	1x 2	Reverse video Input Field Auto Skip Range Check Min Value = Max Value =	Left Justify Copy Value	Integer
EYEAR	5,38	1x 2	Reverse video Input Field Auto Skip Range Check Min Value = Max Value =	Left Justify Copy Value	Integer
STATUS	7,26	1x 1	Reverse video Input Field Auto Skip Range Check Min Value = Max Value =	Left Justify Copy Value	Integer
SCHOOL	9,30	1x40	Reverse video Input Field Auto Skip	Left Justify Copy Value	None

ภาพที่ 6.8 ตัวอย่างรายงานรายละเอียดของจอภาพ STDEDU.SCN (ต่อ)

รายละเอียดรายวิชา		
รหัสวิชา	ชื่อวิชา	จำนวนหน่วยกิต

ศูนย์วิทยทรัพยากร

ภาพที่ 6.9 ตัวอย่างรายงานรายละเอียดของจอภาพ COURSE.SCN

จุฬาลงกรณ์มหาวิทยาลัย

1. Screen Definition

Screen Name : COURSE.SCN

Size of Screen File = 183 bytes

Start Location : 0, 0

End Location : 23,79

No of text field = 1

No of output field = 0

No of input field = 3

Cursor Field Name : -

Message Field Name : -

Previous Screen Name : -

Next Screen Name : -

2. Field Definition

Name	Row,Col	Size	Field Attribute		
HEAD	2,30	1x21	Reverse video Text Field		
COURID	6,11	15x 6	Underline Input Field Scroll	Left Justify	Alphanumeric
COURNM	6,23	15x35	Underline Input Field Scroll	Left Justify	None
COURCR	6,64	15x 2	Underline Input Field Scroll	Left Justify	Integer

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Listing of file : REGISTC.C

```

#include <string.h>
#include <memory.h>

static char *password = {"REGISTER"};

char   passwd[9],           /* password                */
       mess1[36],          /* message field           */
       stdid[8],           /* student id              */
       stdnm[51],          /* student name            */
       sex,                /* sex                     */
       stdad1[51],         /* student address 1      */
       stdad2[51],         /* student address 2      */
       stdad3[51],         /* student address 3      */
       courid[30][7],      /* array of course id     */
       cournm[30][36],     /* array of course name   */
       school[41],        /* old school              */
       zext[4];            /* exit field              */

int    enmenu,             /* menu number             */
       bday, eday,         /* day 1-31                */
       bmon, emon,         /* month 1-12              */
       byear, eyear,       /* year 0-99               */
       status,             /* status of student       */
       courcr[30],         /* array of course credit  */
       rc;                 /* return code             */

main()
{
    int done, done1, i, j;

    rc = 0;

    /* define field */
    scdpcv("vdefine passwd c", 16, &rc, passwd, 9);
    if (rc == 20) {
        printf("screen display is inactive\n");
        exit(0);
    }
    scdpcv("vdefine mess1 c", 17, &rc, mess1, 36);
    scdpcv("vdefine enmenu i", 16, &rc, &enmenu, 2);
    scdpcv("vdefine stdid c", 15, &rc, stdid, 8);
    scdpcv("vdefine stdnm c", 15, &rc, stdnm, 51);
    scdpcv("vdefine sex c", 13, &rc, &sex, 1);
    scdpcv("vdefine bday i", 14, &rc, &bday, 2);
    scdpcv("vdefine eday i", 14, &rc, &eday, 2);
    scdpcv("vdefine bmon i", 14, &rc, &bmon, 2);
    scdpcv("vdefine emon i", 14, &rc, &emon, 2);
    scdpcv("vdefine byear i", 15, &rc, &byear, 2);
    scdpcv("vdefine eyear i", 15, &rc, &eyear, 2);

```

Listing of file : REGISTC.C

```

scdpcv("vdefine stdad1 c", 16, &rc, stdad1, 51);
scdpcv("vdefine stdad2 c", 16, &rc, stdad2, 51);
scdpcv("vdefine stdad3 c", 16, &rc, stdad3, 51);
scdpcv("vdefine school c", 16, &rc, school, 41);
scdpcv("vdefine status i", 16, &rc, &status, 2);
scdpcva("vdefarr courid c", 16, &rc, &courid[0][0], 7, 30);
scdpcva("vdefarr cournm c", 16, &rc, &cournm[0][0], 36, 30);
scdpcva("vdefarr courcr i", 16, &rc, &courcr[0], 2, 30);
scdpcv("vdefine zext c", 14, &rc, zext, 4);

enmenu = 1;
done = 0;
zext[3] = '\0';

while (!done) {
    scdpc("display password", 16, &rc); /* password.scn */
    if (strcmp(zext, "ESC") == 0) { /* if user press Esc key */
        scdpc("clear", 5, &rc); /* clear screen */
        exit(0);
    }
    if (strcmp(password, passwd) != 0) { /* invalid password */
        strcpy(mess1, "INVALID PASSWORD !!! ENTER AGAIN");
        scdpc("sendmsg", 7, &rc); /* send message */
        scdpc("alarm", 5, &rc); /* alarm sound */
        continue;
    }
    done = 1;
}

/* store screen in memory area */

scdpc("sbuffer stdmain", 15, &rc);
scdpc("sbuffer stdrsume", 16, &rc);
scdpc("sbuffer stdedu", 14, &rc);
scdpc("sbuffer course", 14, &rc);

done = 0;
while (!done) {
    scdpc("display stdmain", 15, &rc); /* stdmain.scn */
    if (strcmp(zext, "ESC") == 0) { /* if user press Esc key */
        done = 1;
        continue;
    }
}
switch (enmenu) {
    case 1 : /* student resume */
        donel = 0;
        while (!donel) {
            strcpy(stdid, ""); /* clear all fields */
            strcpy(stdnm, "");
            strcpy(stdad1, "");

```

Listing of file : REGISTC.C

```

        strcpy(stdad2, "");
        strcpy(stdad3, "");
        bday = bmon = byear = 0;
        eday = emon = eyear = 0;
        strcpy(school, "");
        status = 0;

        scdpc("display stdrsume", 16, &rc); /* stdrsume.scn */
        if (strcmp(zext, "ESC") == 0) { /* if user press Esc key */
            donel = 1;
            continue;
        }
        else {
            /*
             * store data to array or file
             */
        }
    }
    break;
case 2 : /* detail of course */
    scdpc("display course", 14, &rc); /* course.scn */
    if (strcmp(zext, "END") == 0) { /* if user press F2 key */
        /*
         * store data to array or file
         */
    }
    break;
default :
    done = 1;
}
}

scdpc("clear", 5, &rc);
}

```

ตัวอย่างโปรแกรมประยุกต์ภาษาฟอร์แทรน

Listing of file : REGISTF.FOR

\$STORAGE:2

PROGRAM REGIST

C

```

INTEGER ENMENU, BDAY, EDAY, BMON, EMON, BYEAR, EYEAR, STATUS
INTEGER COURCR(30)
INTEGER RC, I, DONE, DONE1, LENGH1, LENGH2, LENGH3
CHARACTER*1 SEX
CHARACTER*3 ZEXT
CHARACTER*6 COURID(30)
CHARACTER*7 STDID
CHARACTER*8 PASSWD, PWORD
CHARACTER*35 MESS1, COURNM(30), CONAME(30)
CHARACTER*40 SCHOOL
CHARACTER*50 STDNM, STDAD1, STDAD2, STDAD3
CHARACTER*80 PARM

```

C

DATA PWORD /'REGISTER'/

C

RC = 0

C

C

define field

C

PARM = 'VDEFINE PASSWD C'

LENGH1 = 16

LENGH2 = 8

CALL SCDFV(PARM, LENGH1, RC, PASSWD, LENGH2)

IF (RC .EQ. 20) THEN

OPEN(0, FILE='CON:')

WRITE(0, 'SCREEN DISPLAY IS INACTIVE')

CLOSE(0)

GOTO 8888

ENDIF

PARM = 'VDEFINE MESS1 C'

LENGH1 = 15

LENGH2 = 35

CALL SCDFV(PARM, LENGH1, RC, MESS1, LENGH2)

PARM = 'VDEFINE ENMENU I'

LENGH1 = 16

LENGH2 = 2

CALL SCDFV(PARM, LENGH1, RC, ENMENU, LENGH2)

PARM = 'VDEFINE STDID C'

LENGH1 = 15

LENGH2 = 7

CALL SCDFV(PARM, LENGH1, RC, STDID, LENGH2)

PARM = 'VDEFINE STDNM C'

LENGH1 = 15

LENGH2 = 50

CALL SCDFV(PARM, LENGH1, RC, STDNM, LENGH2)

Listing of file : REGISTF.FOR

```
PARM = 'VDEFINE SEX C'  
LNGTH1 = 13  
LNGTH2 = 1  
CALL SCDFV(PARM, LNGTH1, RC, SEX, LNGTH2)  
PARM = 'VDEFINE BDAY I'  
LNGTH1 = 14  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, BDAY, LNGTH2)  
PARM = 'VDEFINE EDAY I'  
LNGTH1 = 14  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, EDAY, LNGTH2)  
PARM = 'VDEFINE BMON I'  
LNGTH1 = 14  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, BMON, LNGTH2)  
PARM = 'VDEFINE EMON I'  
LNGTH1 = 14  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, EMON, LNGTH2)  
PARM = 'VDEFINE BYEAR I'  
LNGTH1 = 15  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, BYEAR, LNGTH2)  
PARM = 'VDEFINE EYEAR I'  
LNGTH1 = 15  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, EYEAR, LNGTH2)  
PARM = 'VDEFINE STDAD1 C'  
LNGTH1 = 16  
LNGTH2 = 50  
CALL SCDFV(PARM, LNGTH1, RC, STDAD1, LNGTH2)  
PARM = 'VDEFINE STDAD2 C'  
LNGTH1 = 16  
LNGTH2 = 50  
CALL SCDFV(PARM, LNGTH1, RC, STDAD2, LNGTH2)  
PARM = 'VDEFINE STDAD3 C'  
LNGTH1 = 16  
LNGTH2 = 50  
CALL SCDFV(PARM, LNGTH1, RC, STDAD3, LNGTH2)  
PARM = 'VDEFINE STATUS I'  
LNGTH1 = 16  
LNGTH2 = 2  
CALL SCDFV(PARM, LNGTH1, RC, STATUS, LNGTH2)  
PARM = 'VDEFINE SCHOOL C'  
LNGTH1 = 16  
LNGTH2 = 40  
CALL SCDFV(PARM, LNGTH1, RC, SCHOOL, LNGTH2)  
PARM = 'VDEFINE ZEXT C'  
LNGTH1 = 14
```

Listing of file : REGISTF.FOR

```

    LNGTH2 = 3
    CALL SCDFV(PARM, LNGTH1, RC, ZEXT, LNGTH2)
    PARM = 'VDEFARR COURID C'
    LNGTH1 = 16
    LNGTH2 = 6
    LNGTH3 = 15
    CALL SCDFVA(PARM, LNGTH1, RC, COURID, LNGTH2, LNGTH3)
    PARM = 'VDEFARR COURNM C'
    LNGTH1 = 16
    LNGTH2 = 35
    LNGTH3 = 15
    CALL SCDFVA(PARM, LNGTH1, RC, COURNM, LNGTH2, LNGTH3)
    PARM = 'VDEFARR COURCR I'
    LNGTH1 = 16
    LNGTH2 = 2
    LNGTH3 = 15
    CALL SCDFVA(PARM, LNGTH1, RC, COURCR, LNGTH2, LNGTH3)

    ENMENU = 1
    DONE = 0

100 PARM = 'DISPLAY PASSWORD'
    LNGTH1 = 16
    CALL SCDF(PARM, LNGTH1, RC)
    IF (ZEXT .EQ. 'ESC') GOTO 9999
    IF (PASSWD .NE. PSWORD) THEN
        MESS1 = 'INVALID PASSWORD !!! ENTER AGAIN'
        PARM = 'SENDMSG'
        LNGTH1 = 7
        CALL SCDF(PARM, LNGTH1, RC)
        PARM = 'ALARM'
        LNGTH1 = 5
        CALL SCDF(PARM, LNGTH1, RC)
        GOTO 100
    ENDIF

C
C
C
store screen in memory area

    PARM = 'SBUFFER STDMAIN'
    LNGTH1 = 15
    CALL SCDF(PARM, LNGTH1, RC)
    PARM = 'SBUFFER STDRSUME'
    LNGTH1 = 16
    CALL SCDF(PARM, LNGTH1, RC)
    PARM = 'SBUFFER STDEDU'
    LNGTH1 = 14
    CALL SCDF(PARM, LNGTH1, RC)
    PARM = 'SBUFFER COURSE'
    LNGTH1 = 14
    CALL SCDF(PARM, LNGTH1, RC)

```

Listing of file : REGISTF.FOR

```

200 PARM = 'DISPLAY STDMAIN'
    LNGTH1 = 15
    CALL SCDF(PARM, LNGTH1, RC)
    IF (ZEXT .EQ. 'ESC') GOTO 9999

    IF (ENMENU .EQ. 1) THEN
C
C     student resume
C
300  PARM = 'DISPLAY STDRSUME'
    LNGTH1 = 16
    CALL SCDF(PARM, LNGTH1, RC)
    IF (ZEXT .EQ. 'ESC') GOTO 500
C
C     store data to array or file
C
    GOTO 300
ELSE IF (ENMENU .EQ. 2) THEN
C
C     detail of course
C
    STDID = '      '
    STDNM = '      '
    STDAD1 = '      '
    STDAD2 = '      '
    STDAD3 = '      '
    SCHOOL = '      '
    STATUS = 0
    BDAY = 0
    BMON = 0
    BYEAR = 0
    EDAY = 0
    EMON = 0
    EYEAR = 0

400  PARM = 'DISPLAY COURSE'
    LNGTH1 = 14
    CALL SCDF(PARM, LNGTH1, RC)
    IF (ZEXT .EQ. 'ESC') GOTO 500
C
C     store data to array or file
C
    ENDIF
500 GOTO 200
C
C     clear screen and exit
C
9999 PARM = 'CLEAR'
    LNGTH1 = 5

```

Listing of file : REGISTF.FOR

CALL SCDF(PARM, LENGH1, RC)
8888 STOP
END



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอย่างโปรแกรมประยุกต์ภาษาปาสคาล

Listing of file : REGISTP.PAS

```
PROGRAM REGISTER(INPUT,OUTPUT);
```

```
CONST
```

```
    PASSWORD = 'REGISTER';
```

```
LABEL
```

```
    EXIT;
```

```
VAR
```

```
    PARM      : LSTRING(80);      { parameter string      }
    PASSWD    : STRING(8);        { password               }
    MESS1     : STRING(35);       { message field         }
    STDID     : STRING(7);        { student id            }
    STDNM     : STRING(50);       { student name          }
    STDAD1    : STRING(50);       { student address 1     }
    STDAD2    : STRING(50);       { student address 2     }
    STDAD3    : STRING(50);       { student address 3     }
    SCHOOL    : STRING(40);       { old school             }
    ZEXT      : STRING(3);        { exit field            }
    SEX       : BYTE;             { sex                    }
    ENMENU,   { menu number      }
    BDAY,     { day of birth date }
    EDAY,     { day of entrance date }
    BMON,     { month of birth date }
    EMON,     { month of entrance date }
    BYEAR,   { year of birth date }
    EYEAR,   { year of entrance date }
    STATUS,  { student status }
    RC,      { return code }
    I, J,
    DONE,
    DONE1    : INTEGER;
    COURID   : ARRAY [1..30] OF STRING(6); { array of course id }
    COURNM   : ARRAY [1..30] OF STRING(35); { array of course name }
    COURCR   : ARRAY [1..30] OF INTEGER;   { array of course credit }
```

```
PROCEDURE SCPPP ( { general interface }
    VAR PARM      : LSTRING;
    VAR RC        : INTEGER
); EXTERN;
```

```
PROCEDURE SCPPV ( { vdefine interface }
    VAR PARM      : LSTRING;
    VAR RC        : INTEGER;
    VNAME         : ADSMEM;
    LENGTH        : WORD
); EXTERN;
```

```
PROCEDURE SCPPVA ( { vdefarr interface }
    VAR PARM      : LSTRING;
    VAR RC        : INTEGER;
    VNAME         : ADSMEM;
```

Listing of file : REGISTP.PAS

```

        LENGTH : WORD;
        SIZE   : WORD
        ); EXTERN;

BEGIN
    RC := 0;

    { define field }

    PARM := 'VDEFINE ZEXT C';
    SCDPPV(PARM, RC, ADS ZEXT, 3);

    IF (RC = 20) THEN
    BEGIN
        WRITELN ('SCREEN DISPLAY IS INACTIVE');
        GOTO EXIT;
    END;

    PARM := 'VDEFINE PASSWD C';
    SCDPPV(PARM, RC, ADS PASSWD, 8);
    PARM := 'VDEFINE MESS1 C';
    SCDPPV(PARM, RC, ADS MESS1, 35);
    PARM := 'VDEFINE ENMENU I';
    SCDPPV(PARM, RC, ADS ENMENU, 2);
    PARM := 'VDEFINE STDID C';
    SCDPPV(PARM, RC, ADS STDID, 7);
    PARM := 'VDEFINE STDNM C';
    SCDPPV(PARM, RC, ADS STDNM, 50);
    PARM := 'VDEFINE SEX C';
    SCDPPV(PARM, RC, ADS SEX, 1);
    PARM := 'VDEFINE BDAY I';
    SCDPPV(PARM, RC, ADS BDAY, 2);
    PARM := 'VDEFINE EDAY I';
    SCDPPV(PARM, RC, ADS EDAY, 2);
    PARM := 'VDEFINE BMON I';
    SCDPPV(PARM, RC, ADS BMON, 2);
    PARM := 'VDEFINE EMON I';
    SCDPPV(PARM, RC, ADS EMON, 2);
    PARM := 'VDEFINE BYEAR I';
    SCDPPV(PARM, RC, ADS BYEAR, 2);
    PARM := 'VDEFINE EYEAR I';
    SCDPPV(PARM, RC, ADS EYEAR, 2);
    PARM := 'VDEFINE STDAD1 C';
    SCDPPV(PARM, RC, ADS STDAD1, 50);
    PARM := 'VDEFINE STDAD2 C';
    SCDPPV(PARM, RC, ADS STDAD2, 50);
    PARM := 'VDEFINE STDAD3 C';
    SCDPPV(PARM, RC, ADS STDAD3, 50);
    PARM := 'VDEFINE SCHOOL C';
    SCDPPV(PARM, RC, ADS SCHOOL, 40);

```

Listing of file : REGISTP.PAS

```

    PARM := 'VDEFINE STATUS I';
    SCDPPV(PARM, RC, ADS STATUS, 2);
    PARM := 'VDEFARR COURID C';
    SCDPPVA(PARM, RC, ADS COURID, 6, 15);
    PARM := 'VDEFARR COURNM C';
    SCDPPVA(PARM, RC, ADS COURNM, 35, 15);
    PARM := 'VDEFARR COURCR I';
    SCDPPVA(PARM, RC, ADS COURCR, 2, 15);

    ENMENU := 1;
    DONE := 0;

    WHILE DONE = 0 DO
    BEGIN
        PARM := 'DISPLAY PASSWORD'; { password.scn }
        SCDPP(PARM, RC);
        IF (ZEXT = 'ESC') THEN      { if user press Esc key }
        BEGIN
            PARM := 'CLEAR';        { clear screen }
            SCDPP(PARM, RC);
            RETURN;
        END;
        IF (PASSWD <> PASSWORD) THEN { invalid password }
        BEGIN
            MESS1 := 'INVALID PASSWORD !!! ENTER AGAIN ';
            PARM := 'SENDMSG';      { send message }
            SCDPP(PARM, RC);
            PARM := 'ALARM';        { alarm sound }
            SCDPP(PARM, RC);
        END
        ELSE
            DONE := 1;
    END;

    { .store screen to memory area }

    PARM := 'SBUFFER STDMAIN';
    SCDPP(PARM, RC);
    PARM := 'SBUFFER STDRSUME';
    SCDPP(PARM, RC);
    PARM := 'SBUFFER STDEDU';
    SCDPP(PARM, RC);
    PARM := 'SBUFFER COURSE';
    SCDPP(PARM, RC);

    DONE := 0;
    WHILE DONE = 0 DO
    BEGIN
        PARM := 'DISPLAY STDMAIN'; { stdmain.scn }
        SCDPP(PARM, RC);
    
```

Listing of file : REGISTP.PAS

```

IF (ZEXT = 'ESC') THEN          { if user press Esc key }
BEGIN
    PARM := 'CLEAR';           { clear screen }
    SCDPP(PARM, RC);
    RETURN;
END;

CASE ENMENU OF
  1 : BEGIN                      { student resume }
      DONE1 := 0;
      WHILE DONE1 = 0 DO
          BEGIN

              { clear all fields }

              STDID := '      ';
              STDNM := '      ';
              STDAD1 := '      ';
              STDAD2 := '      ';
              STDAD3 := '      ';
              SCHOOL := '      ';
              BDAY := 0;
              BMON := 0;
              BYEAR := 0;
              EDAY := 0;
              EMON := 0;
              EYEAR := 0;
              STATUS := 0;

              PARM := 'DISPLAY STDRSUME'; { stdrsume.scn }
              SCDPP(PARM, RC); { if user press Esc key }
              IF (ZEXT = 'ESC') THEN
                  BEGIN
                      DONE1 := 1;
                      CYCLE;
                  END
              ELSE
                  { store data to array or file }
                  ;
              END;
          END;
      2 : BEGIN                  { detail of course }
          PARM := 'DISPLAY COURSE'; { course.scn }
          SCDPP(PARM, RC);
          IF (ZEXT = 'END') THEN  { if user press F2 key }
              { store data to array or file }
              ;
          END
      OTHERWISE
          DONE := 1;
  
```


Listing of file : REGISTP.PAS

END;
END;
EXIT :
END.



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย