

ใบปฏิบัติการการเขียนโปรแกรมภาษาซี

การทดลองที่ 10

การเขียนโปรแกรมโครงสร้างข้อมูล

วัตถุประสงค์

- เพื่อให้ผู้อ่านสามารถเขียน โปรแกรมการใช้โครงสร้างได้
- เพื่อให้ผู้อ่านมีทักษะในการเขียน โปรแกรมการใช้ยูเนียนได้
- เพื่อให้ผู้อ่านสามารถเขียน โปรแกรมการโดยนำทฤษฎีมาประยุกต์ใช้
ในการเขียนโปรแกรมได้

ความรู้พื้นฐาน

การทดลองการเขียนโปรแกรมภาษาซี

การเขียนโปรแกรมด้วยโครงสร้างข้อมูล (Alexander, A., Online) (Byron S. Gottfried, 1990) เป็นการเก็บข้อมูลของตัวแปร ที่มีลักษณะชนิดข้อมูลที่แตกต่างกัน แต่สามารถนำมาเป็นกลุ่มข้อมูลชุดเดียวกันได้ เพื่อเป็นประโยชน์ต่อการจัดการข้อมูลที่เป็นชุดหน่วยความจำให้เหมาะสมในด้านการจองหน่วยความจำ

(Kenneth A.Barclay., 1990) และการนำโครงสร้างมาสร้างในการเป็นฐานข้อมูลขนาดที่เหมาะสมกับงานที่มีขนาดไม่ใหญ่มาก ก็สามารถใช้การเขียนโปรแกรมด้วยโครงสร้างมาช่วยแก้ปัญหาได้ (M.Tim Grady, 1980) เพื่อให้เกิดความเข้าใจอย่างประจักษ์ ด้วยการทำการทดลอง และให้ผู้อ่านสังเกตผลการทำงานของแต่ละโปรแกรม ให้ผู้อ่านปฏิบัติดังนี้

สำนักพิมพ์ศูนย์ส่งเสริมวิชาการ

โจทย์ที่ 1. ให้ผู้อ่านเขียนโปรแกรมที่กำหนด และให้ทำการคอมไพล์ให้ได้ผลการ
ทำงานของโปรแกรมห่างตัวอย่างพร้อมทั้งอธิบายผลการทำงานของโปรแกรม

```
#include <stdio.h>
#include <conio.h>

#define MAX_X 200 /* จุดสูงสุดในแนวแกน x ของรูปสี่เหลี่ยม */
#define MAX_Y 300 /* จุดสูงสุดในแนวแกน y ของรูปสี่เหลี่ยม */

#define YES 1
#define NO 0

struct point {
    int x;
    int y;
};

struct rectangle {
    struct point pt1;
    struct point pt2;
};

/* ฟังก์ชันเพื่อใช้ในการกำหนดจุดเริ่มต้นของสี่เหลี่ยมให้กับโครงสร้าง */
struct point makepoint(int x, int y) {
    struct point temp;
    temp.x = x;
    temp.y = y;
    return ( temp );
}
```

```

/* ฟังก์ชันเพื่อใช้ในการบวกค่า x และ y ของจุด 2 จุด */
struct point addpoint(struct point p1, struct point p2) {
    p1.x += p2.x;
    p1.y += p2.y;
    return ( p1);
}
/* ฟังก์ชันเพื่อใช้ตรวจสอบว่าจุดของในรูปสี่เหลี่ยมหรือไม่ */
/* ถ้าไม่อยู่จะคืนค่า 0 ถ้าอยู่จะคืนค่าที่ไม่ใช่ 0 */
int pinrect(struct point p, struct rectangle r) {
return ( p.x >= r.pt1.x && p.x <= r.pt2.x && p.y >= r.pt1.y && p.y
<= r.pt2.y );
}
void main() {
    struct rectangle screen;
    struct point middle, sumPoint, specPoint;
    int ans;
    clrscr();
    screen.pt1 = makepoint(20,30); /* กำหนดจุดแรกของสี่เหลี่ยม */
    screen.pt2 = makepoint(MAX_X, MAX_Y); /* กำหนดจุดที่สอง */
    printf("\nLower bound : %d, %d", screen.pt1.x, screen.pt1.y);
    printf("\nUpper bound : %d, %d", screen.pt2.x, screen.pt2.y);
}

```

```

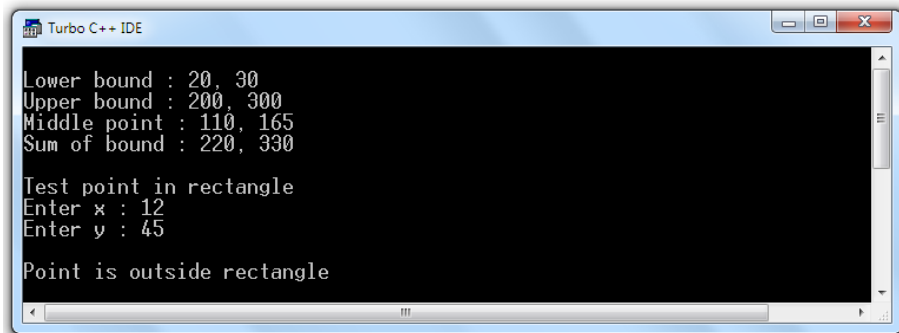
/* หาจุดกึ่งกลางของรูปสี่เหลี่ยม */
    middle = makepoint((screen.pt1.x+screen.pt2.x)/2,
(screen.pt1.y+screen.pt2.y)/2);
    printf("\nMiddle point : %d, %d", middle.x, middle.y);

/* หาผลบวกระหว่างจุดแรกและจุดที่ 2 ของสี่เหลี่ยม*/
    sumPoint = addpoint(screen.pt1, screen.pt2);
    printf("\nSum of bound : %d, %d", sumPoint.x, sumPoint.y);

/* รับค่าจุดและทดสอบว่าจุดนั้นอยู่ในรูปสี่เหลี่ยมที่ระบุหรือไม่ */
    printf("\n\nTest point in rectangle");
    printf("\nEnter x : ");
    scanf("%d", &(specPoint.x));
    printf("Enter y : ");
    scanf("%d", &(specPoint.y));
    ans = pinrect(specPoint, screen);
    if (ans == YES)
        printf("\nPoint is in rectangle");
    else
        printf("\nPoint is outside rectangle");
    getch();
}

```

ผลการทำงานโปรแกรม



```

Turbo C++ IDE
Lower bound : 20, 30
Upper bound : 200, 300
Middle point : 110, 165
Sum of bound : 220, 330

Test point in rectangle
Enter x : 12
Enter y : 45

Point is outside rectangle
  
```

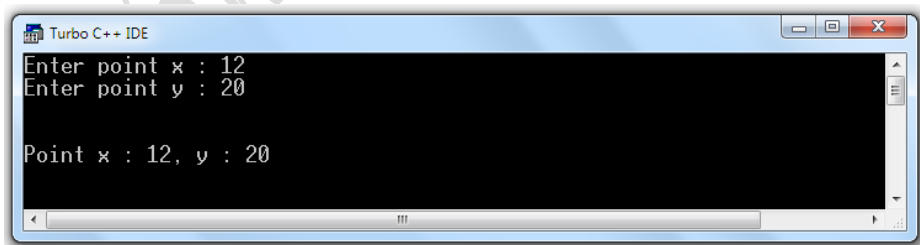
โจทย์ที่ 2. ให้ผู้อ่านเขียนโปรแกรมด้านล่าง และทำการเขียนผลการทำงานโปรแกรมพร้อมอธิบายเหตุผลในการแสดงผลการแสดงผลมาพอสังเขป (แสดงการใช้งานตัวแปรพอยน์เตอร์)

```

#include <stdio.h>
#include <conio.h>
struct point {
int x;   int y;
};
void readPoint(struct point *);
void printPoint(struct point *);
void main() { clrscr();
struct point pt;
readPoint(&pt);
printPoint(&pt);      }
  
```

```
void readPoint(struct point *pPt) {  
    printf("Enter point x : ");  
    scanf("%d", &pPt->x); /* scanf("%d", &(*pPt).x); */  
    printf("Enter point y : ");  
    scanf("%d", &pPt->y); /* scanf("%d", &(*pPt).y); */  
}  
  
void printPoint(struct point *pP) {  
    printf("\n\nPoint x : %d, y : %d", pP->x, pP->y);  
    /* printf("\n\nPoint x : %d, y : %d", (*pP).x, (*pP).y); */  
    getch();  
}
```

ผลการทำงานโปรแกรม



The screenshot shows the Turbo C++ IDE window. The output window displays the following text:

```
Enter point x : 12  
Enter point y : 20  
  
Point x : 12, y : 20
```

โจทย์ที่ 3. ให้ผู้อ่านเขียนโปรแกรมด้านล่าง และทำการเขียนผลการทำงานโปรแกรมพร้อมอธิบายเหตุผลในการแสดงผลการแสดงผลมาพอสังเขป

```
#include <stdio.h>
#include <conio.h>
#define NO_SCORE 5
typedef struct {
    char name[16]; char surname[20];
    int score[NO_SCORE]; float total;
} Student;
void readStudentData(Student *);
void findTotalScore(Student *);
float findAverage(Student);
void findLessThanTen(Student);
void main() { clrscr();
    Student std;
    float avg;
    readStudentData(&std);
    findTotalScore(&std);
    avg = findAverage(std);
    printf("\n\nAverage score is %.2f", avg);
    findLessThanTen(std);
    getch();
}
```



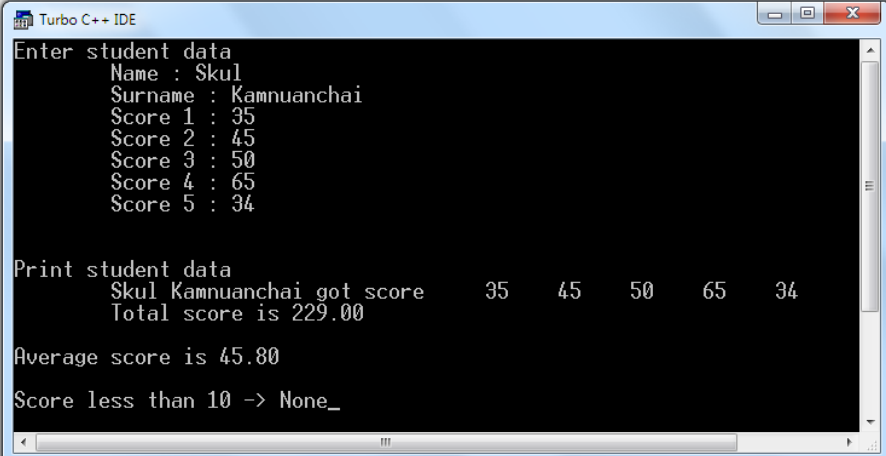
```
void readStudentData(Student *pStd) {
    int i;
    printf("Enter student data\n");
    printf("\tName : ");
    scanf("%s", &pStd->name);
    printf("\tSurname : ");
    scanf("%s", &pStd->surname);
    for (i=0; i<NO_SCORE; i++) {
        printf("\tScore %d : ", i+1);
        scanf("%d", &pStd->score[i]);
    }
}

void findTotalScore(Student *pStd) {
    int i;
    printf("\n\nPrint student data");
    printf("\n\t%s %s got score ", pStd->name, pStd->surname);
    pStd->total = 0.0;
    for (i=0; i<NO_SCORE; i++) {
        printf("%6d", pStd->score[i]);
        pStd->total += pStd->score[i];    }
    printf("\n\tTotal score is %.2f", pStd->total);
}
```

```
float findAverage(Student s) {
    return(s.total/NO_SCORE);
}

void findLessThanTen(Student s) {
    int i,count=0;
    printf("\n\nScore less than 10");
    for (i=0; i<NO_SCORE; i++) {
        if (s.score[i] < 10) {
            printf("\n\tTest no.%d - %d", i+1, s.score[i]);
            count++;
        }
    }
    if (count==0) /* กรณีที่ไม่มีทดสอบครั้งใดได้น้อยกว่า 10 */
        printf(" -> None");
}
```

บันทึกผลการทำงานโปรแกรม



```
Turbo C++ IDE
Enter student data
  Name : Skul
  Surname : Kamnuanchai
  Score 1 : 35
  Score 2 : 45
  Score 3 : 50
  Score 4 : 65
  Score 5 : 34

Print student data
  Skul Kamnuanchai got score    35    45    50    65    34
  Total score is 229.00

Average score is 45.80

Score less than 10 -> None_
```

โจทย์ที่ 4. ให้ผู้อ่านเขียนโปรแกรมด้านล่าง และอธิบายการทำงานของโปรแกรม มาพอเข้าใจพร้อมเขียนผลการทำงานของโปรแกรม

```
#include <stdio.h>
#include <conio.h>
#define MAX_STUDENT 4
typedef struct {
    char name[16];
    char surname[20];
    int age;
} Student;

void readStudent(Student []);
void printStudent(Student []);
float findAverageAge(Student []);
void LessThanAverage(Student [], float);
void findMaxMinAge(Student [], int *, int *);
void printMaxAge(Student [], int);
void printMinAge(Student [], int);
void main() { clrscr();
    Student student[MAX_STUDENT];
    float avg;
    int maxAge, minAge;
    readStudent(student);
    printStudent(student);
```

```
avg = findAverageAge(student);
    printf("\n\nAverage age is %.2f", avg);
    LessThanAverage(student, avg);
    findMaxMinAge(student, &maxAge, &minAge);
    printMaxAge(student, maxAge);
    printMinAge(student, minAge);
    getch();
}

void readStudent(Student stdarr[]) {
    int i;
    printf("Enter student data");
    for (i=0; i<MAX_STUDENT; i++) {
        printf("\n\tNo.%d", i+1);
        printf("\n\tName : ");
        scanf("%s", stdarr[i].name);
        printf("\tSurname : ");
        scanf("%s", stdarr[i].surname);
        printf("\tAge : ");
        scanf("%d", &stdarr[i].age);
    }
}
```

```

void printStudent(Student stdarr[]) {
    int i;    printf("\n\nStudent data");
    for (i=0; i<MAX_STUDENT; i++) {
        printf("\n\tNo.%d", i+1);
        printf("\tName : %16s", stdarr[i].name);
        printf("\tSurname : %20s", stdarr[i].surname);
        printf("\tAge : %d", stdarr[i].age);
    }
}

float findAverageAge(Student stdarr[]) {
    float sumAge=0.0, avgAge;    int i;
    for (i=0; i<MAX_STUDENT; i++) {
        sumAge = sumAge + stdarr[i].age;
    }
    avgAge = sumAge / MAX_STUDENT;
    return(avgAge);
}

void LessThanAverage(Student stdarr[], float avg) {
    int i;
    printf("\n\nLess than average age %.2f", avg);
    for (i=0; i<MAX_STUDENT; i++) {
        if (stdarr[i].age < avg)
            printf("\n%s %s with %d years old", stdarr[i].name,
                stdarr[i].surname, stdarr[i].age);
    }
}

```

```
void findMaxMinAge(Student stdarr[], int *max, int *min) {
    int i, maxIdx, minIdx;
    maxIdx = minIdx = 0;
    for (i=1; i<MAX_STUDENT; i++) {
        if (stdarr[i].age > stdarr[maxIdx].age)
            maxIdx = i;
        else if (stdarr[i].age < stdarr[minIdx].age)
            minIdx = i;
    }
    *max = stdarr[maxIdx].age;
    *min = stdarr[minIdx].age;
}

void printMaxAge(Student stdarr[], int maxAge) {
    int i;
    printf("\n\nMaximum age report");
    printf("\nMaximum age is %d", maxAge);
    for (i=0; i<MAX_STUDENT; i++) {
        if (stdarr[i].age == maxAge)
            printf("\n\t%s %s", stdarr[i].name, stdarr[i].surname);
    }
}
```

```
void printMinAge(Student stdarr[], int minAge) {  
    int i;  
    printf("\n\nMinimum age report");  
    printf("\nMinimum age is %d", minAge);  
    for (i=0; i<MAX_STUDENT; i++) {  
        if (stdarr[i].age == minAge)  
            printf("\n\t%s %s", stdarr[i].name, stdarr[i].surname);  
    }  
}
```

อุปกรณ์

1. เครื่องคอมพิวเตอร์
2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
3. ตัวโปรแกรมภาษาซี หรือ
4. ตัวโปรแกรม CodeBlock
5. หน่วยความจำเคลื่อนที่ (Handy Drive, External Drive)

วิธีการทดลอง

1. ให้ผู้อ่านเรียกโปรแกรมที่ใช้ในการเขียนโปรแกรมภาษาซี
 - 1.1 โปรแกรมภาษาซีจาก Turbo C
 - 1.2 โปรแกรม CodeBlock
 2. ให้ผู้อ่านฝึกการ โปรแกรมที่ผู้อ่านได้เลือกทำการติดตั้งในเครื่องคอมพิวเตอร์
 - 2.1 ทดสอบการเขียนโค้ด
 - 2.2 ทดสอบการสั่งให้โปรแกรมทำการแปลคำสั่ง (Compiler)
 - 2.3 ทดลองทำการสั่งให้โปรแกรมเริ่มทำงาน (Run)
 3. ให้ผู้อ่านทดสอบหาจุดผิดพลาดของโปรแกรมและทำการแก้ไขจุดผิดพลาดโดยยึดทฤษฎีมาเป็นหลักในการแก้ปัญหา
 4. ให้ผู้อ่านได้ทดลองทำการป้อนตามที่เนื้อหาได้กำหนด ทั้งส่วนทฤษฎีและการทดลอง ให้ผลการทำงานโปรแกรม ตรงตามที่เนื้อหาที่กำหนด
 5. ให้ผู้อ่านได้ทำการเขียนผลการทดลองตามกำหนด
-