

## CS102 การบ้าน 4

### 1. (25 คะแนน) Grade Report (File & Struct)

จงเขียนโปรแกรมภาษา C ที่ประกอบด้วยฟังก์ชันต่างๆ ที่ทำงานตามที่กำหนด โดยอาจเขียนฟังก์ชันอื่นเพิ่มได้ตามที่นักศึกษาเห็นสมควร

กำหนดให้โปรแกรมรับข้อมูลจากแฟ้มข้อมูล myClassCS007.txt ซึ่งเก็บข้อมูลของนักศึกษาที่ลงทะเบียนเรียนวิชา CS007 โดยวิชานี้แต่ละปีจะมีโควตา รับคนลงทะเบียนปีละไม่เกิน 25 คน ในแฟ้มข้อมูลนี้จะมีข้อมูลแบ่งเป็นสองส่วน ส่วนแรกคือบรรทัดแรก เป็นจำนวนนักศึกษาทั้งหมดที่ลงทะเบียนในปีนั้นๆ ส่วนที่สองคือบรรทัดที่สองเป็นต้นไป แต่ละบรรทัดจะเป็นข้อมูลของนักศึกษาแต่ละคน

โดยข้อมูลของนักศึกษาแต่ละคนประกอบด้วย เลขทะเบียน ชื่อ นามสกุล คะแนนสอบเก็บคะแนนทั้งหมด 5 ครั้ง แต่ละครั้งมีคะแนนเต็ม 20 คะแนน

ตัวอย่างของข้อมูลภายในแฟ้มข้อมูล myClassCS007.txt ในปีนี้ เป็นดังนี้

|            |       |       |       |       |       |      |       |  |
|------------|-------|-------|-------|-------|-------|------|-------|--|
| 6          |       |       |       |       |       |      |       |  |
| 5809610010 | Alice | Brown | 15.5  | 9.8   | 10.25 | 10   | 18.75 |  |
| 5809610022 | Bill  | Green | 5.5   | 19.8  | 20    | 16.3 | 8.75  |  |
| 5809610037 | Cathy | Blue  | 17.25 | 10.75 | 5.75  | 9.1  | 16.2  |  |
| 5809610045 | David | Grey  | 12    | 7.8   | 0     | 13.5 | 16.75 |  |
| 5809610059 | Elle  | Pink  | 15.5  | 19.8  | 20    | 16.3 | 8.75  |  |
| 5809610063 | Fred  | White | 10    | 13.5  | 9.75  | 0    | 0     |  |

ให้นักศึกษาเขียนโปรแกรมเพื่อรับข้อมูลเข้า ส่วนที่เป็นข้อมูลของนักศึกษาทั้งหมด จัดเก็บลงในอาร์เรย์ของสตรัคที่มีโครงสร้างดังต่อไปนี้ (โดยเก็บข้อมูลลงในฟิลด์ที่เกี่ยวข้องข้อสี่ฟิลด์แรก)

```
struct StudentInfo {  
    char ID[11];  
    char firstN[16];  
    char lastN[16];  
    float test[5];  
    float totalScore;  
    char grade;  
}
```

จากนั้น ให้โปรแกรมคำนวณคะแนนรวมและเกรดของนักศึกษาแต่ละคนโดยใช้เกณฑ์ดังนี้

คะแนนรวม [90,100] ได้เกรด A,

คะแนนรวม [80,90) ได้เกรด B,

คะแนนรวม [70, 80) ได้เกรด C,

คะแนนรวม [60, 70) ได้เกรด D, และ

คะแนนรวม [0, 60) ได้เกรด F

แล้วจัดเก็บค่าคะแนนรวมและเกรดลงในฟิลด์ totalScore และฟิลด์ grade ตามลำดับ

โปรแกรมประกอบด้วยฟังก์ชัน main( ) ซึ่งจะทำการแสดงเมนูทางเลือก สี่ทางเลือก ดังนี้

Menu:

1. Show a student data
2. Print the grade report on the screen
3. Print the grade report in a file called grade.out
0. Quit program

Enter your choice <1, 2, 3 or 0> :

แล้วรอรับค่าตัวเลือกจากผู้ใช้ หากผู้ใช้ป้อนผิด (ไม่ใช่ 1, 2, 3 หรือ 0) ให้ขึ้นข้อความเตือนว่า Invalid choice และรอรับตัวเลือกใหม่จนกว่าจะถูก

- ถ้าผู้ใช้เลือกเมนู 1 ให้เรียกใช้ฟังก์ชันชื่อ studentInfo() โดยส่งอาร์เรย์ของสตริงที่ไปเป็นพารามิเตอร์ โดยฟังก์ชันนี้จะขึ้นข้อความ

Please enter your ID:

เพื่อถามเลขทะเบียนของนักศึกษา จากนั้นรอรับเลขทะเบียนเข้ามา แล้วแสดงข้อมูลของนักศึกษาคนนั้นออกมาที่หน้าจอ แล้วกลับไปแสดงเมนูอีก

ตัวอย่างเช่น สมมติผู้ใช้ป้อนเลขทะเบียนเป็น

5809610059

จะได้ผลลัพธ์ว่า

5809610059 Elle Pink 80.35 B

- ถ้าผู้ใช้เลือกเมนู 2 ให้ 1 ให้เรียกใช้ฟังก์ชันชื่อ gradeReportOnScreen() โดยส่งอาร์เรย์ของสตริงที่ไปเป็นพารามิเตอร์ โดยฟังก์ชันนี้จะทำการแสดงรายงานเกรดของนักศึกษา โดยให้มีข้อมูลดังต่อไปนี้ เลขทะเบียน ชื่อ นามสกุล คะแนนรวม และเกรดที่นักศึกษาแต่ละคนได้รับ และสรุปคะแนนสูงสุด ต่ำสุด และคะแนนเฉลี่ยของทั้งชั้นด้วย จากนั้นกลับไปแสดงเมนูอีก

- ถ้าผู้ใช้เลือกเมนู 3 ให้โปรแกรมเรียกใช้ฟังก์ชันชื่อ gradeReportInFile() โดยส่งอาร์เรย์ของสตริงที่ไปเป็นพารามิเตอร์ โดยฟังก์ชันนี้จะทำการสร้างแฟ้มข้อมูล grade.out เพื่อเก็บผลลัพธ์รายงานเกรดของนักศึกษา โดยให้มีข้อมูลดังต่อไปนี้ เลขทะเบียน ชื่อ นามสกุล คะแนนรวม และเกรดที่นักศึกษาแต่ละคนได้รับ และสรุปคะแนนสูงสุด ต่ำสุด และคะแนนเฉลี่ยของทั้งชั้น แล้วกลับไปแสดงเมนูอีก

- ถ้าผู้ใช้เลือกเมนู 0 ให้จบโปรแกรม

ตัวอย่างของผลลัพธ์ที่โปรแกรมของนักศึกษาต้องแสดงออกทางหน้าจอ (ถ้าเลือกเมนู 2) หรือ ต้องเขียนลงในแฟ้มข้อมูลผลลัพธ์ชื่อ grade.out (ถ้าเลือกเมนู 3) เป็นดังนี้

```
Grade Report:
5809610010 Alice Brown      64.30      D
5809610022 Bill  Green      70.35      C
5809610037 Cathy Blue       59.05      F
5809610045 David Grey       50.05      F
5809610059 Elle  Pink       80.35      B
5809610063 Fred  White      33.25      F
Maximum Score: 80.35
Minimum Score: 33.25
Average Score: 59.56
```

## 2.(20 คะแนน) Magic Squares (2D-array)

จงเขียนโปรแกรมภาษา C ซึ่งประกอบด้วยฟังก์ชัน 2 ฟังก์ชัน คือ ฟังก์ชัน main() และ ฟังก์ชัน isMagic() ซึ่งทำหน้าที่ดังต่อไปนี้

- 1) ฟังก์ชัน main() แสดงข้อความให้ผู้ใช้งานป้อนขนาดของตารางสี่เหลี่ยมจัตุรัสขนาดตั้งแต่ 3-9 ถ้าผู้ใช้งานป้อนข้อมูลไม่ถูกต้องให้แสดงข้อความ “Invalid Input!!” จากนั้นให้เข้าไปทำงานขั้นตอนที่ 3)
- 2) ถ้าผู้ใช้งานป้อนขนาดตารางอยู่ในช่วง 3-9 ได้ถูกต้องแล้ว ให้แสดงข้อความบอกผู้ใช้งานให้ป้อนตัวเลขตั้งแต่ 1 ถึง ขนาดตาราง<sup>2</sup> (เช่นถ้าผู้ใช้งานป้อนขนาดตารางเป็น 4 ให้ผู้ใช้งานป้อนเลข 1 ถึง 16) ในลักษณะ ขนาดตาราง \* ขนาดตาราง ทั้งนี้ กำหนดให้เก็บข้อมูลตารางนี้ในรูปของอาร์เรย์สองมิติ (2D-array)
- 3) จากนั้นให้ฟังก์ชัน main() เรียกใช้ฟังก์ชัน isMagic() โดยส่งตารางเลขนั้นไปเป็นพารามิเตอร์ (และอาจมีพารามิเตอร์อื่นอีก ตามที่นักศึกษาเห็นสมควร) ฟังก์ชันนี้จะพิจารณาว่าตารางที่ถูกส่งมานั้นเป็น Magic Square หรือไม่ โดยตารางจะเป็น Magic Square ก็ต่อเมื่อเลขทุกตัวในตารางมีค่าไม่ซ้ำกัน และผลรวมของตัวเลขในแถว ทุกคอลัมน์ในตารางมีค่าเท่ากัน และมีค่าเท่ากับผลรวมตัวเลขในแนวทแยงมุมทั้งสองแนวด้วย เรียกค่านี้ว่า Magic Constant ซึ่งฟังก์ชัน isMagic() จะส่งค่า Magic Constant กลับมา หากตารางนั้นเป็น Magic Square แต่ถ้าไม่ใช่ก็จะส่งค่ากลับเป็น 0
- 4) ฟังก์ชัน main() แสดงผลบอกว่าตารางที่ป้อนเข้ามาเป็น Magic Square หรือไม่ ถ้าใช่จะบอกด้วยว่าผลรวมในทุกแนว (Magic Constant) ได้เท่าไร

### ตัวอย่างหน้าจอ output

**Please enter size of the square (3-9):** 1 <enter>  
**Invalid Input!!**

**Please enter size of the square (3-9):** 5 <enter>  
**Enter numbers 1 to 25 into your square:**

|    |    |    |    |    |
|----|----|----|----|----|
| 1  | 4  | 12 | 11 | 3  |
| 15 | 13 | 9  | 7  | 2  |
| 21 | 22 | 23 | 24 | 25 |
| 6  | 8  | 5  | 9  | 10 |
| 14 | 16 | 17 | 18 | 19 |

**Your square is not a magic square!!**

**Try more <y/n>?** y <enter>

**Please enter size of the square (3-9):** -5 <enter>  
**Invalid Input!!**

**Please enter size of the square (3-9):** 3 <enter>  
**Enter numbers 1 to 9 into your square:**

|   |   |   |
|---|---|---|
| 4 | 9 | 2 |
| 3 | 5 | 7 |
| 8 | 1 | 6 |

**Congratulations!! Your square is a magic square.**  
**The magic constant of this square is 15.**

**Try more <y/n>?** Y <enter>

**Please enter size of the square (3-9):** 3 <enter>  
**Enter numbers 1 to 9 into your square:**

|   |   |   |
|---|---|---|
| 5 | 5 | 5 |
| 5 | 5 | 5 |
| 5 | 5 | 5 |

**Your square is not a magic square!!**

**Try more <y/n>?** n <enter>

**Bye.**

<จบโปรแกรม>

### 3. (5 คะแนน) isPalindrome (recursive function) :

จากการบ้าน#3 เราได้เขียนโปรแกรมในการตรวจสอบว่าชุดของตัวอักษรเป็น palindrome (คำที่สะกดเหมือนกับการสะกดกลับหลังของคำนั้น) หรือไม่ ในการบ้านนี้ กำหนดให้นักศึกษาเขียนฟังก์ชัน isPalindrome() แต่กำหนดให้เป็น **recursive function** (ฟังก์ชันที่มีการเรียกใช้ตัวมันเอง) โดยฟังก์ชันนี้จะ return ค่ากลับเป็น 1 ถ้าคำที่รับผ่านทาง parameter เป็น palindrome และ return ค่ากลับเป็น 0 ถ้าไม่ใช่

เมื่อนำฟังก์ชันดังกล่าวมาเขียนร่วมกับ main() ที่กำหนดให้ต่อไปนี้

```
int main() {
    char word[25];
    printf ("Please enter a word: ");
    scanf ("%s", word);
    if (isPalindrome(word))
        printf ("%s is a palinedrome.\n", word);
    else printf ("%s is not a palinedrome.\n", word);
    return 0;
}
```

โปรแกรมจะสามารถทำงานได้ดังตัวอย่างต่อไปนี้

ตัวอย่างหน้าจอผลลัพธ์ (รันครั้งเดียว)

```
Please enter a word: MODEL <enter>
MODEL is not a palindrome.
Would you like to continue? <y/n> y <enter>
Please enter a word: RACECAR <enter>
RACECAR is a palindrome.
Would you like to continue? <y/n> n <enter>
<จบโปรแกรม>
```

☺ 🍀 🍀 🍀 GOOD LUCK 🍀 📖 🖋️ 🕒