

ข้อกำหนด: ให้นักศึกษาเขียนโปรแกรมภาษาซีตามที่โจทย์กำหนด แล้วบันทึกเป็นไฟล์ .c ให้ตั้งชื่อว่า cs102Hw3-x-y เมื่อ x คือเลขข้อ และ y คือเลขทะเบียนของนักศึกษา (ตัวอย่าง นักศึกษาเลขทะเบียน 6809670000 จะต้องตั้งชื่อไฟล์คำตอบข้อสอบของการบ้าน 3 ข้อที่ 1 ว่า cs102Hw3-1-6809670000.c) แล้วอัปโหลดไฟล์ขึ้นกล่องส่งในคอร์สเว็บให้ตรงข้อ ไม่เกินกำหนดส่ง

กำหนดส่ง: การบ้านครั้งที่ 3 ข้อ 1-4 ไม่เกิน 23:55 ของวันศุกร์ที่ 31 ตุลาคม 2568

การบ้านครั้งที่ 3 ข้อ 5 ไม่เกิน 23:55 ของวันจันทร์ที่ 3 พฤศจิกายน 2568

- ให้นักศึกษาเขียนโปรแกรมภาษาซีเพื่อรับค่าเลขจำนวนเต็ม 1 จำนวนคือ N คือจำนวนของตัวเลขที่จะรับเข้ามาตามด้วยชุดตัวเลขจำนวนจริงอีก N ตัว (โดยเลขแต่ละจำนวนคั่นด้วยช่องว่าง) เก็บค่าชุดตัวเลขเอาไว้ในตัวแปรประเภทย่อยจากข้อมูลชุดตัวเลขดังกล่าวให้โปรแกรมที่นักศึกษาเขียนคำนวณและ/หรือแสดงค่าต่อไปนี้
 - ค่าต่ำสุด ตำแหน่งของค่าต่ำสุด ค่าสูงสุด ตำแหน่งของค่าต่ำสุด และค่าเฉลี่ย ตามลำดับ
 - พิมพ์ค่าในลำดับย้อนกลับ (Reverse order)
 - พิมพ์ค่าทุกค่าที่มีค่ามากกว่าค่าเฉลี่ย (Above average) ของของชุดตัวเลขทั้งหมด
 - พิมพ์ทุกคู่อันดับของตัวเลขทุกตัวที่สร้างจากตัวเลขในอาเรย์ที่เป็นค่าติดลบทั้งคู่

หมายเหตุ:

- นักศึกษาต้องใช้อาร์เรย์ในการแก้ปัญหาในข้อนี้และการเข้าถึงค่าของสมาชิกของอาร์เรย์นักศึกษาต้องใช้ `[]` หรือ `Subscript` เพื่อระบุตำแหน่งที่ต้องการเข้าถึงข้อมูลในตัวแปรอาร์เรย์ มิเช่นนั้น โปรแกรมของนักศึกษาจะไม่ได้รับการตรวจให้คะแนน
- N เป็นเลขจำนวนเต็มบวกมีค่าไม่เกิน 50
- ตำแหน่งของค่าต่ำสุดให้ตอบตำแหน่งที่อยู่ใกล้ 0 มากที่สุด
- ในการพิมพ์ค่าเลขจำนวนจริง ให้พิมพ์ด้วยความละเอียดเพียง 2 ตำแหน่ง

ตัวอย่างการรันโปรแกรม#1

```
5
1.5 -1.0 -7.5 2.0 -3.0
Maximum value:      2.00
Minimum value:      -7.50
Maximum index:       3
Minimum index:       2
Average:             -1.60
Reverse order: -3.00 2.00 -7.50 -1.00 1.50
Above average list: 2.00 -1.00 1.50
Negative coordinates: (-1.00, -7.50), (-1.00, -3.00), (-7.50, -3.00),
```

ตัวอย่างการรันโปรแกรม#2

```
3
1.0 -2.0 -1.5
Maximum value:      1.00
Minimum value:      -2.00
Maximum index:       0
Minimum index:       1
Average:             -0.83
Reverse order: -1.50 -2.00 1.00
Above average list: 1.00
Negative coordinates: (-2.00, -1.50),
```

2. ให้นักศึกษาเขียนโปรแกรมภาษาซีเพื่อรับค่า String เข้ามาหนึ่งค่า โดยที่ความยาวของ String ไม่เกิน 26 อักขระ โดย String ดังกล่าวเป็นลำดับตัวอักษรที่อาจจะไม่ได้ถูกเรียงลำดับตามตัวหนังสือ และต้องถูกสร้างด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ A-Z เท่านั้น อย่างไรก็ตามใน String ดังกล่าวอาจจะมีตัวอักษรที่ซ้ำกันและหายไป ให้คำนวณหาตัวอักษรที่หายไป และตัวอักษรที่เกิดการซ้ำกันทุกตัวใน String ที่ทำให้ String ดังกล่าวไม่สามารถมีอักษรที่สามารถนำมาเรียงกันได้ตั้งแต่ A...X ได้ โดย x คืออักษรตัวสุดท้ายที่ไกลจาก A ที่สุดเท่าที่เป็นไปได้จากความยาวของ String ที่รับเข้ามา ในกรณีที่ String มีตัวอักษรเรียงครบตั้งแต่ A...X ให้แสดงข้อความว่า NO MISSING ในกรณีที่มีตัวอักษรที่ซ้ำกันและตัวอักษรที่หายไป ให้แสดงตัวอักษรดังกล่าวตามลำดับ (เรียงตามตัวอักษร)

ตัวอย่างที่ 1

String ที่รับเข้ามาคือ ECABD จะเห็นว่า String มีความยาว 5 ตัวอักษร และประกอบไปด้วยอักษร A B C D E (A...E) จะเห็นว่า E คือตัวอักษรที่ไกลที่สุดจาก A มากที่สุดเท่าที่เป็นไปได้จาก String ที่มีความยาว 5 ตัวอักษร แต่ปรากฏว่าใน String ดังกล่าวไม่มีอักษรใดซ้ำกันเลย และเรียงลำดับถูกต้องจาก A ถึง E จึงถือว่าไม่มีตัวอักษรใดที่ขาดหายไปด้วย

ตัวอย่างการรันโปรแกรม#1

ECABD

NO MISSING

ตัวอย่างที่ 2

String ที่รับเข้ามาคือ AAABD จะเห็นว่า String มีความยาว 5 ตัวอักษร และประกอบไปด้วยอักษร A A A B D (A...E) จะเห็นว่า E คือตัวอักษรที่ไกลที่สุดจาก A มากที่สุดเท่าที่เป็นไปได้ แต่ใน String ดังกล่าวมีตัวอักษร A ซ้ำกันถึง 3 ตัวอักษร ทำให้เกิดอักษรที่หายไปทั้งหมด 2 ตัวอักษรที่จะทำให้ตัวอักษรทั้งหมดใน String มีครบทั้ง 5 ตัวอักษรที่ไม่ซ้ำกันเลยตั้งแต่ A จนถึง E คือตัวอักษร B C D ก็คือ A (ตัวอักษรที่ซ้ำกัน) และ CE (ตัวอักษรที่หายไป)

ตัวอย่างการรันโปรแกรม#2

AAABD

A

CE

3. ให้นักศึกษาเขียนโปรแกรมภาษาซีเพื่อรับค่าเลขจำนวนเต็ม 1 จำนวนคือ N คือขนาดของ Array ($3 \leq N \leq 8$) และใน N บรรทัดถัดมา ในแต่ละบรรทัดให้รับค่าจำนวนเต็มอีก N จำนวน กันด้วยช่องว่าง เมื่อรับมาครบ N บรรทัดแล้วนักศึกษาจะสังเกตว่าตัวเลขที่รับเข้ามาจะเป็น Matrix จตุรัสขนาด $N \times N$ และให้นักศึกษาตรวจสอบว่า Matrix ดังกล่าวเป็น Magic square หรือไม่ โดย Magic square จะต้องมีความสมบัติทุกประการต่อไปนี้
- ตัวเลขที่อยู่ในอาร์เรย์ต้องเป็นตัวเลขที่แตกต่างกันในช่วง 1 จนถึง N^2
 - ผลรวมของทุกแถว ทุกหลัก และแนวทแยงจะต้องมีค่าเท่ากันทั้งหมด

ตัวอย่างการรันโปรแกรม#1

3

2 7 6

9 5 1

4 3 8

This is a magic square!

ตัวอย่างการรันโปรแกรม#2

3

3 2 1

3 5 1

7 3 1

This is NOT a magic square!

4. ให้นักศึกษาปรับโปรแกรมในข้อ 1. โดยให้นักศึกษาใช้ตัวแปร Pointers สำหรับแทนการเข้าถึงสมาชิกของอาร์เรย์แทนการใช้ Subscript

ตัวอย่างการรับโปรแกรม (เหมือนข้อ 1. ทุกประการ)

5. ให้นักศึกษาปรับปรุงโปรแกรมในข้อ 3 (Magic square) โดยให้นักศึกษาโหลดไฟล์ CS102Hw3_5_XXXXXXXXXX.c ที่จัดเตรียมไว้ในคอร์สเว็บและให้**นิยามฟังก์ชันเพิ่มเติมในไฟล์ดังกล่าว 2 ฟังก์ชัน**ตามข้อกำหนดด้านล่าง เพื่อให้ฟังก์ชัน main() ในไฟล์ที่กำหนดให้ทำงานได้อย่างถูกต้อง
- a. ฟังก์ชัน **readSquare** รับพารามิเตอร์ 2 ตัว:
 - (1) อาร์เรย์ A เป็นอาร์เรย์ 2 มิติ ขนาด 8×8
 - (2) N เป็นขนาดของ Input squareตามลำดับ และรับค่าตัวเลข $N \times N$ ตัว มาเก็บค่าเอาไว้ในอาร์เรย์ A โดยจะต้องเก็บให้อยู่ในรูปของ Matrix จัดจรัส โดยฟังก์ชันนี้**ไม่ต้องคืนค่า**
 - b. ฟังก์ชัน **findMagicNumber** รับพารามิเตอร์ 2 ตัว:
 - (1) อาร์เรย์ A เป็นอาร์เรย์ 2 มิติ ขนาด 8×8
 - (2) N เป็นขนาดของ Input squareตามลำดับ โดยฟังก์ชันนี้จะทำการตรวจสอบว่า อาร์เรย์ A เป็น Magic square หรือไม่
 - หาก**ใช่**ให้คืนค่าผลรวมที่มีค่าเท่ากับตามข้อกำหนดของ Magic square ที่ระบุไว้ในโจทย์ข้อ 3.
 - หาก**ไม่ใช่**ให้คืนค่า -1กำหนดให้ Return type ของฟังก์ชันนี้เป็นชนิด `int`

หมายเหตุ:

1. นักศึกษาต้องเขียนฟังก์ชันตามข้อกำหนดที่ระบุไว้ใน a. และ b.
2. นักศึกษาห้ามเปลี่ยนแปลงการทำงานของฟังก์ชัน `main()` โดยเด็ดขาด
3. หากโปรแกรมของนักศึกษาไม่เป็นไปตามข้อ 1. และ 2. การบ้านข้อนี้จะไม่ได้รับการตรวจให้คะแนน
4. ก่อนส่งให้นักศึกษาเปลี่ยนชื่อไฟล์ CS102Hw3 5 xxxxxxxxxxx.c โดยแทน xxxxxxxxxx ด้วยเลขทะเบียนนักศึกษา

ตัวอย่างการรันโปรแกรม #1

```
4
8 5 3 3
1 6 5 1
7 2 7 4
4 6 1 9
Your square is NOT a magic square.
```

ตัวอย่างการรันโปรแกรม #2

```
3
6 1 8
7 5 3
2 9 4
Your square is a magic square!! with the magic constant = 15.
```