

## แบบฝึกหัดปฏิบัติการคาบที่ 10: Structure

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564 Section.....

1. จุดในระนาบสามารถที่จะแสดงได้โดยการใช้ระบบ Coordinate x และ y ดังนั้นเราสามารถเขียนจุดในระนาบได้โดยการใช้ตัวแปรแบบโครงสร้างที่มีสองฟิลด์ดังแสดงด้านล่าง

typedef struct

```
{
    int x;
    int y;
}POINT
```

```
#include<stdio.h>
#include<math.h>
typedef struct Point{
    float x;
```

จงเขียนโปรแกรมเพื่อทำการรับค่าข้อมูลแบบโครงสร้างของจุด (POINT) แล้วทำการเรียกฟังก์ชันเพื่อทำการคำนวณหา ระยะทางระหว่างจุดสองจุดโดยใช้ระยะทางแบบยูคลิดีเนียน ซึ่งมีนิยามการทำงานดังนี้

```
float Dist(struct Point p1,struct Point p2){
```

Dist (Point1, Point 2) =  $\sqrt{(Point1.x - Point2.x)^2 + (Point1.y - Point2.y)^2}$

หลังจากนั้นให้เรียกฟังก์ชันเพื่อระบุหาว่าจุดทั้งสองอยู่ Quadrant ที่เท่าไร

```
distance = sqrt(pow(p1.x - p2.x,2)+pow(p1.y - p2.y,2));
return distance;
```

**ข้อมูลอินพุต**

บรรทัดแรกเป็นจำนวนจุด n (1<=n<=100)  
n บรรทัดถัดไปเป็นตำแหน่งของจุดในพิกัด x และ y (-1000<x, y<1000)

```
}
int quadrant(struct Point p1){
```

**ข้อมูลเอาต์พุต**

n บรรทัดแรกเป็นผลลัพธ์ของ Quadrant มีค่าตั้งแต่ 1- 4  
บรรทัดสุดท้ายเป็นผลลัพธ์ของระยะทาง

```
q = 3;
}else if (p1.x < 0 && p1.y < 0){
    q = 3;
}else if (p1.x < 0 && p1.y > 0){
    q = 2;
}return q;
```

**ตัวอย่าง**

อินพุต	เอาต์พุต
2 2 -2 2 2	4 1 4

```
int main(){
    int n,i,k,l=0;
    scanf("%i",&n);
    Point p11[n];
    float d[100];
    int q[n];
```

2. Structure ชื่อ vector3D เป็นเวกเตอร์สามมิติ มีสมาชิกเป็นเลขทศนิยมสามตัวคือ X, Y, Z จงเขียนโปรแกรมเพื่อรับค่า ทั้งสามตัวของเวกเตอร์ จากนั้นคำนวณหาผลบวกของเวกเตอร์โดยใช้ฟังก์ชัน FindLength ความยาวของเวกเตอร์คำนวณได้โดย  $\sqrt{X^2 + Y^2 + Z^2}$

```
for (i=0;i<n;i++){
    scanf("%f%f%f",&p11[i].x,&p11[i].y,&p11[i].z);
    q[i] = quadrant(p11[i]);
}
for (i=0;i<n;i++){
    printf("%i\n",q[i]);
}
```

**ข้อมูลอินพุต**

บรรทัดแรกเป็นจำนวนจุด n (1<=n<=100)  
n บรรทัดถัดไปเป็นตำแหน่งของจุดในพิกัด x,y และ z (-1000<x, y, z<1000)

```
for (i=0;i<n-1;i++){
    for(k=1+i;k<n;k++){
        d[i] = Dist(p11[i],p11[k]);
        printf("%.0f\n",d[i]);
        l++;
    }
}
```

**ข้อมูลเอาต์พุต**

บรรทัดสุดท้ายเป็นผลลัพธ์

```
return 0;
}
```

**ตัวอย่าง**

อินพุต	เอาต์พุต
4 0.5 1.2 2.0	2.39

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564.....ตอนเรียน Lab ที่.....

3. ที่ร้านสะดวกซื้อแห่งหนึ่งเมื่อทำการรับเงินจากลูกค้าจะทำการแยกเงินแต่ละราคาใส่ไว้ในช่องเก็บเงินที่ประกอบด้วยชนิดของเงินแต่ละราคา คือ 1000, 500, 100, 50, 20, 10, และ 1 บาท จงเขียนโปรแกรมเพื่อที่จะรับจำนวนเงินจากลูกค้าเพื่อส่งไปยังฟังก์ชันที่ทำหน้าที่คำนวณหาจำนวนเงินแต่ละชนิดราคาหลังจากนั้นคืนค่าตัวแปรโครงสร้างที่ประกอบด้วยช่องเก็บเงินแต่ละชนิดราคา คือ 1000, 500, 100, 50, 20, 10, และ 1 บาท ตามลำดับ

#### ข้อมูลอินพุท

บรรทัดแรกเป็นจำนวนเงินจากลูกค้า ( $0 < a < 10000000$ )

#### ข้อมูลเอาต์พุท

บรรทัดสุดท้ายเป็นผลลัพธ์โครงสร้างที่ประกอบด้วยช่องเก็บเงินแต่ละชนิดราคา คือ 1000, 500, 100, 50, 20, 10, และ 1 บาท ตามลำดับ

#### ตัวอย่าง

อินพุท	เอาต์พุท
1751	1 1 2 1 0 0 1

4. ในไพ่สำรับหนึ่งประกอบด้วย face values และ suits

โดยที่ face values ประกอบด้วย A, 2, 3, 4, 5, 6, 7, 8, 9, J, Q, K

ส่วน suits ประกอบด้วย โพดำ (Spade) ♠

โพแดง หรือ หัวใจ (Heart) ♥

ข้าวหลามตัด (Diamond) ♦

ดอกจิก (Club) ♣

จงเขียนโปรแกรมเพื่อกำหนดโครงสร้างของไพ่ (Deck) ที่ประกอบไปด้วย faces และ suits หลังจากนั้นผู้ใช้กำหนดรายละเอียดของไพ่ตามจำนวนที่ผู้ใช้กำหนด หลังจากนั้นให้โปรแกรมทำการเรียงลำดับไพ่ทั้ง 4 ใบดังกล่าวจากน้อยไปหามากแสดงผลพร้อมทั้งหาค่าผลรวมของไพ่ที่บ่อนเข้ามา

#### ข้อมูลอินพุท

บรรทัดแรกเป็นจำนวนไพ่ทั้ง n ใบ ( $1 \leq n \leq 52$ )

n บรรทัดต่อไปแสดงรายละเอียดของไพ่ตามที่ผู้ใช้กำหนด

#### ข้อมูลเอาต์พุท

บรรทัดต่อไปแสดงการเรียงลำดับไพ่ทั้ง n ใบดังกล่าวจากน้อยไปหามาก

บรรทัดสุดท้ายแสดงผลรวมของไพ่ที่บ่อนเข้ามา

#### ตัวอย่าง

อินพุท	เอาต์พุท
3 3-C 7 S A H	A-H, 3-C, 7-S 11

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564.....ตอนเรียน Lab ที่.....

5. โครงสร้างข้อมูลแบบ Stack ประกอบด้วย ~~ตัวแปรอาร์เรย์ที่ใช้ในการเก็บค่าของ Stack ขนาด N สมาชิก และตัวแปร Top~~ สำหรับชี้ค่าบนสุดของ Stack โดยตัวแปร Top จะใช้ประกอบการเพิ่มและลบข้อมูลที่อยู่ใน Stack

~~การเพิ่มหรือลบข้อมูลจะทำให้ได้ค่าเฉพาะข้อมูลที่อยู่บนสุดของ Stack~~

~~ฟังก์ชันพื้นฐานของ Stack คือ~~

1. ~~ฟังก์ชัน Push เป็นฟังก์ชันสำหรับเพิ่มข้อมูลเข้าไปใน stack~~

2. ~~ฟังก์ชัน Pop เป็นฟังก์ชันสำหรับดึงข้อมูลที่อยู่บนสุดออกจาก stack~~

~~จงเขียนโปรแกรมเพื่อจำลองการทำงานของ Stack โดยเมื่อผู้ใช้ต้องการ Push ให้พิมพ์ P เพื่อเพิ่มข้อมูล และถ้าผู้ใช้ต้องการ Pop ให้พิมพ์ X โดยโปรแกรมจะดึงข้อมูลที่อยู่บนสุดออกมา~~

~~ตัวอย่าง~~

~~Please select operation: P~~

~~Please input data: 20~~

~~Please select operation: P~~

~~Please input data: 15~~

~~Please select operation: P~~

~~Please input data: 35~~

~~Please select operation: P~~

~~Please input data: 10~~

~~Please select operation: X~~

~~Data is : 10~~

~~Please select operation: X~~

~~Data is : 35~~