

ใบงานการทดลองที่ 14
เรื่อง เทคนิคการหาค่ามากที่สุดและน้อยที่สุด

1. จุดประสงค์ทั่วไป

3.1. รู้และเข้าใจแนวทางการประยุกต์การเขียนโปรแกรมคอมพิวเตอร์

2. จุดประสงค์เชิงพฤติกรรม

3.1.1. ออกแบบแนวทางการแก้ไข้ปัญหาเพื่อหาค่ามากที่สุดและน้อยที่สุด

3.1.2. วิเคราะห์แนวทางการหาค่ามากที่สุดและน้อยที่สุด

3.1.3. วางหลักการโครงสร้างการหาค่ามากที่สุดและน้อยที่สุด

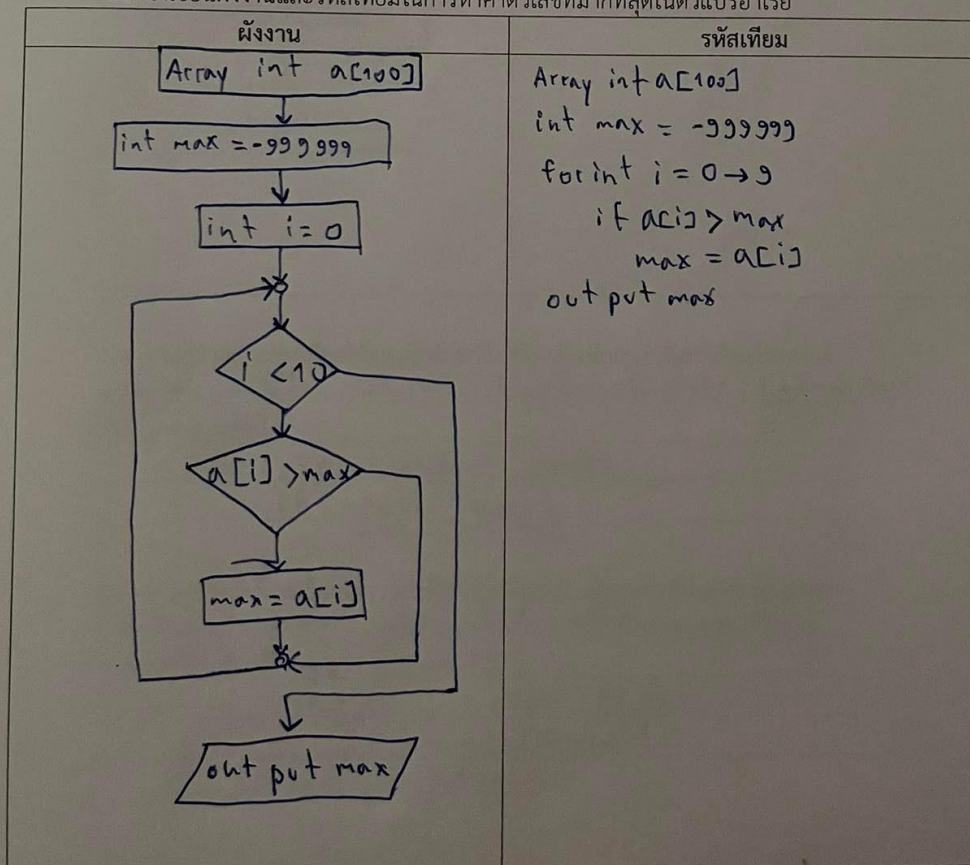
3.1.4. ฝึกหัดและทดลองการหาค่ามากที่สุดและน้อยที่สุด

3. เครื่องมือและอุปกรณ์

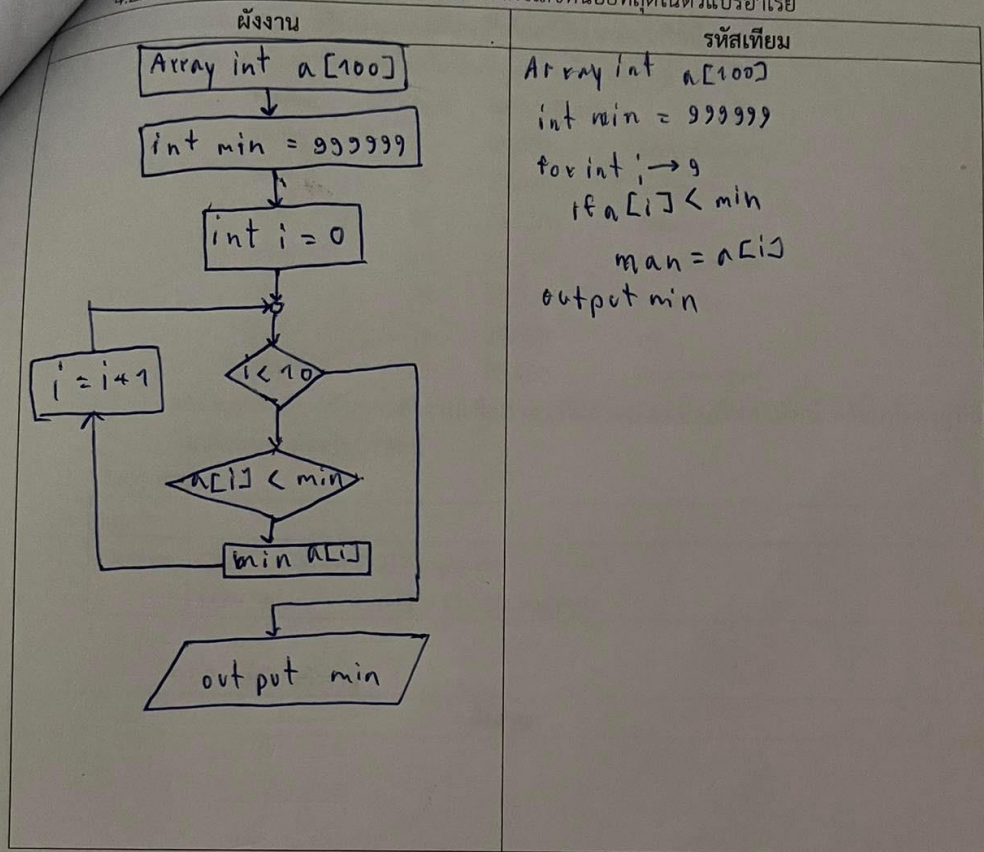
เครื่องคอมพิวเตอร์ 1 เครื่องที่ติดตั้งโปรแกรม Dev-C

4. ทฤษฎีการทดลอง

4.1. จงเขียนผังงานและรหัสเทียมในการหาค่าตัวเลขที่มากที่สุดในตัวแปรอาร์เรย์



4.2. จงเขียนผังงานและรหัสเทียมในการหาค่าตัวเลขที่น้อยที่สุดในตัวแปรอาร์เรย์



4.3. จงยกตัวอย่างการเรียกใช้งานคำสั่งในฟังก์ชันหลักเพื่อหาค่าที่มากที่สุดในอาร์เรย์
 คำสั่ง for เพื่อใช้ในการวนรอบจนกว่าข้อมูลเป็นค่าเดียว แล้วใช้ if เช็คค่าข้อมูล
 นั้นมากกว่าหรือไม่

```

for (i = 0; i < count; i++) {
    if (data[i] > max) {
        max = data[i];
    }
}

```

4.4. จงยกตัวอย่างการเรียกใช้งานคำสั่งในฟังก์ชันหลักเพื่อหาค่าที่น้อยที่สุดในอาร์เรย์
 เช่นเดียวกันกับกรณีหาค่ามากที่สุดคือ คำสั่ง for เพื่อใช้ในการวนรอบจนกว่า
 ข้อมูลเป็นค่าเดียว แล้วใช้ if เช็คค่าข้อมูลนั้นน้อยกว่าหรือไม่

```

for (i = 0; i < count; i++) {
    if (data[i] < min) {
        min = data[i];
    }
}

```


ระดับการปฏิบัติงาน

5.1. จงเขียนผังงานและโค้ดโปรแกรมเพื่อแก้ไขโจทย์ปัญหาดังต่อไปนี้

5.1.1. จงบันทึกข้อมูลต่อไปนี้ลงในโครงสร้างข้อมูลแบบอาร์เรย์

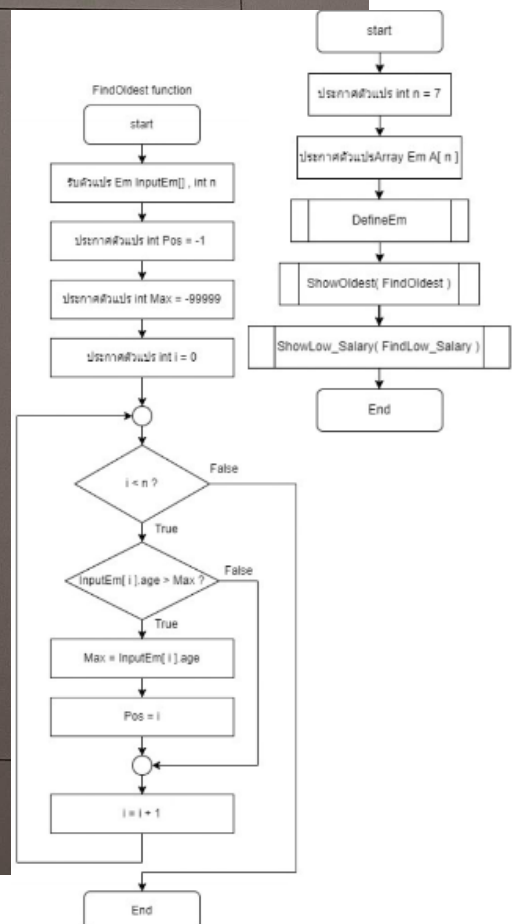
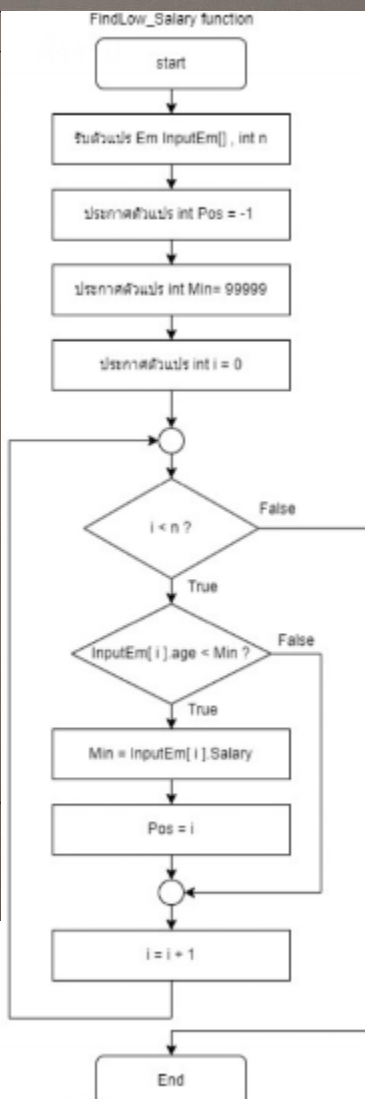
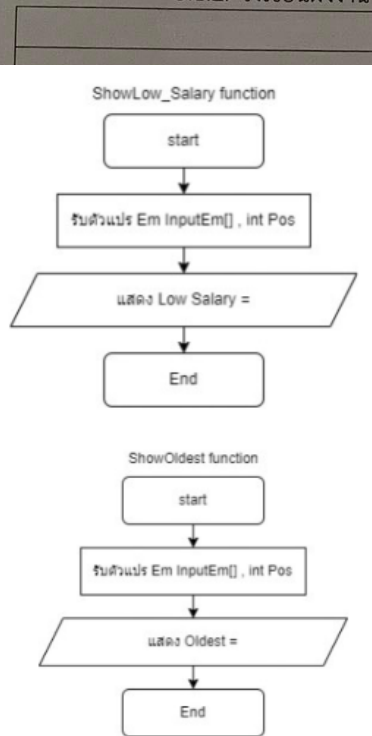
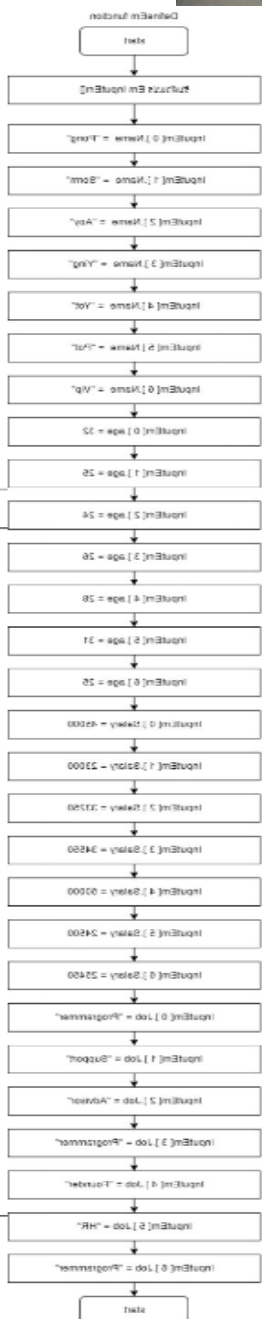
Name	Age	Salary(Bath)	Job
Pong	32	45,000	Programmer
Som	25	23,000	Support
Aoy	24	33,250	Advisor
Ying	26	34,550	Programmer
Yot	28	50,000	Founder
Pot	31	24,500	HR
Vip	25	25,450	Programmer

และจงสร้างฟังก์ชันการทำงานเพื่อหาคนที่อายุน้อยที่สุดในบริษัทนี้ พร้อมทั้งหาผู้ที่มีเงินเดือนน้อยที่สุดในบริษัทนี้

Test case

Input	
Output	Oldest = Pong (32) Programmer Low Salary = Som (25) 23,000 Support

5.1.2. จงเขียนผังงาน



```

1 #include<stdio.h>
2 #include<string.h>
3
4 struct Employees {
5     char Name[ 20 ] ;
6     int age ;
7     int Salary ;
8     char Job [ 20 ] ;
9 } typedef Em ;
10
11 void DefineEm( Em[] ) ;
12 int FindOldest(Em[] , int) ;
13 int FindLow_Salary(Em[] , int) ;
14 void ShowOldest(Em[] , int) ;
15 void ShowLow_Salary(Em[] , int) ;
16
17 int main(){
18     int n = 7 ;
19     Em A[ n ] ;
20     DefineEm( A ) ;
21
22     ShowOldest( A , FindOldest( A , n ) ) ;
23     ShowLow_Salary( A , FindLow_Salary( A , n ) ) ;
24
25     return 0 ;
26 } // end function
27
28 void DefineEm( Em InputEm[] ){
29     strcpy( InputEm[ 0 ].Name , "Pong" ) ;
30     strcpy( InputEm[ 1 ].Name , "Sorm" ) ;
31     strcpy( InputEm[ 2 ].Name , "Aoy" ) ;
32     strcpy( InputEm[ 3 ].Name , "Ying" ) ;
33     strcpy( InputEm[ 4 ].Name , "Yot" ) ;
34     strcpy( InputEm[ 5 ].Name , "Pot" ) ;
35     strcpy( InputEm[ 6 ].Name , "Vip" ) ;
36
37     InputEm[ 0 ].age = 32 ;
38     InputEm[ 1 ].age = 25 ;
39     InputEm[ 2 ].age = 24 ;
40     InputEm[ 3 ].age = 26 ;
41     InputEm[ 4 ].age = 28 ;
42     InputEm[ 5 ].age = 31 ;
43     InputEm[ 6 ].age = 25 ;
44
45     InputEm[ 0 ].Salary = 45000 ;
46     InputEm[ 1 ].Salary = 23000 ;
47     InputEm[ 2 ].Salary = 33250 ;
48     InputEm[ 3 ].Salary = 34550 ;
49     InputEm[ 4 ].Salary = 50000 ;
50     InputEm[ 5 ].Salary = 24500 ;
51     InputEm[ 6 ].Salary = 25450 ;
52
53     strcpy( InputEm[ 0 ].Job , "Programmer" ) ;
54     strcpy( InputEm[ 1 ].Job , "Support" ) ;
55     strcpy( InputEm[ 2 ].Job , "Advisor" ) ;
56     strcpy( InputEm[ 3 ].Job , "Programmer" ) ;
57     strcpy( InputEm[ 4 ].Job , "Founder" ) ;
58     strcpy( InputEm[ 5 ].Job , "HR" ) ;
59     strcpy( InputEm[ 6 ].Job , "Programmer" ) ;
60 } // end function
61
62 int FindOldest(Em InputEm[] , int n ){
63     int Pos = -1 ;
64     int Max = -99999 ;
65     for( int i = 0 ; i < n ; i++ ){
66         if( InputEm[ i ].age > Max ){
67             Max = InputEm[ i ].age ;
68             Pos = i ;
69         } //end if
70     } // end for
71     return Pos ;
72 } // function
73
74 int FindLow_Salary(Em InputEm[] , int n ){
75     int Pos = -1 ;
76     int Min = 99999 ;
77     for( int i = 0 ; i < n ; i++ ){
78         if( InputEm[ i ].age < Min ){
79             Min = InputEm[ i ].Salary ;
80             Pos = i ;
81         } //end if
82     } // end for
83     return Pos ;
84 } // end function
85
86 void ShowOldest(Em InputEm[] , int Pos){
87     printf( "Oldest = %s (%2d) %3d %10s\n" , InputEm[ Pos ].Name , InputEm[ Pos ].age , InputEm[ Pos ].Salary , InputEm[ Pos ].Job ) ;
88 } // end function
89
90 void ShowLow_Salary(Em InputEm[] , int Pos){
91     printf( "Low Salary = %s (%2d) %3d %10s\n" , InputEm[ Pos ].Name , InputEm[ Pos ].age , InputEm[ Pos ].Salary , InputEm[ Pos ].Job ) ;
92 } // end function

```

แกรม

โค้ดโปรแกรม

สรุปผลการปฏิบัติงาน

การปรับปรุงงาน พบว่า สามารถคำนวณวงเงินหักภาษีได้ไม่ structure

มาเพิ่มเพิ่มเพื่อหา เงินได้เพิ่มหักที่หักหรือหักสูงสุด 1x for ในขณนำขอมว

มาเพิ่มเพิ่มใน ic เป็นข้อใด

7. คำถามทางการทดลอง

7.1. จงอธิบายเหตุผลของการกำหนดค่าเริ่มต้นตัวแปร $\text{max} = -999999$

เพื่อให้ค่าเริ่มต้นของตัวแปร Max จะไม่ใช่ค่าที่เฉพาะเจาะจงเกินไป

7.2. จงระบุความแตกต่างในการหาค่ามากที่สุด/น้อยที่สุดของตัวแปรอาเรย์แบบธรรมดาและตัวแปรอาเรย์ของโครงสร้างข้อมูล

ตัวอย่าง $\text{Max} = -99999$ และ $\text{Min} = 99999$ และให้ตัวแปรอาเรย์ที่เก็บค่าไว้ไม่เปลี่ยนแปลง

Input Em Ci age > max

Input Em Ci age < min