

ใบงานการทดลองที่ 16
เรื่อง เทคนิคการค้นหาและการเรียงข้อมูลภายในตัวแปรอาร์เรย์

1. จุดประสงค์ทั่วไป

1.1. ให้เข้าใจแนวทางการประยุกต์การเขียนโปรแกรมคอมพิวเตอร์

2. จุดประสงค์เชิงพฤติกรรม

- 3.1.10. ออกแบบแนวทางการค้นหาข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.11. วิเคราะห์แนวทางการค้นหาข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.12. วางหลักการโครงสร้างการค้นหาข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.13. ฝึกหัดและทดลองการค้นหาข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.14. แก้ไขและประยุกต์การค้นหาข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.15. ออกแบบแนวทางการจัดเรียงข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.16. วิเคราะห์แนวทางการจัดเรียงข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.17. วางหลักการโครงสร้างการจัดเรียงข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.18. ฝึกหัดและทดลองการจัดเรียงข้อมูลภายในตัวแปรชนิดอาร์เรย์
- 3.1.19. แก้ไขและประยุกต์การจัดเรียงข้อมูลภายในตัวแปรชนิดอาร์เรย์

3. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่องที่ติดตั้งโปรแกรม Dev-C

4. ทฤษฎีการทดลอง

4.1. จงบอกและอธิบายหลักการทำงานของการทำงานของการค้นหาข้อมูลแบบเรียงลำดับ

บทที่ 10 อธิบายหลักการทำงานของการทำงานของการค้นหาข้อมูลแบบเรียงลำดับ
ที่ได้อธิบายไว้ก่อนหน้านี้แล้วคือแบบที่หนึ่งที่เราได้เรียนมาคือแบบที่หนึ่งที่เราได้เรียนมา
คือแบบที่หนึ่งที่เราได้เรียนมาคือแบบที่หนึ่งที่เราได้เรียนมาคือแบบที่หนึ่งที่เราได้เรียนมา
คือแบบที่หนึ่งที่เราได้เรียนมาคือแบบที่หนึ่งที่เราได้เรียนมาคือแบบที่หนึ่งที่เราได้เรียนมา

4.2. จงวาดภาพประกอบการทำงานในข้อที่ 4.1 มาอย่างละเอียด

ภาพประกอบการทำงานอย่างละเอียด

$i=0$ $i \rightarrow 5$ $find = a_x$

a_1	a_2	a_3	a_4	a_5
-------	-------	-------	-------	-------

Start \rightarrow if $a[i] = a_x$
out put $a[i]$

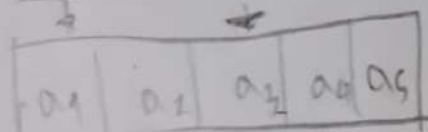
4.3 จงบอกและอธิบายหลักการทำงานของเรียงลำดับข้อมูลแบบเลือก
 การทำงานของข้อมูล เพื่อเปรียบเทียบค่าของข้อมูลที่จะใส่
 แล้วใส่ลงในตัวแปร หรือ pointer เพื่อเก็บไว้แสดง

4.4 จงวาดภาพประกอบการทำงานในข้อที่ 4.3 มาอย่างละเอียด

ภาพประกอบการทำงานอย่างละเอียด

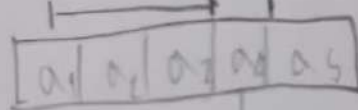
วิธีแรก

for $i = 1 \rightarrow j$ (น้อยไปมาก)



if $a[i] < a[j]$

do:



temp = $a[j]$

for $j = 1 \rightarrow 5$

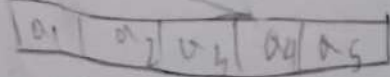
then: $a1, a2, a3, a4, a5$

เป็นการเรียงโดย
 กิ่งเล็กแล้วเก็บ
 เล็กไปเรื่อยๆ
 ถึงค่าที่เล็กใน
 กลุ่มที่เลือก

temp
 Finally \rightarrow break for i

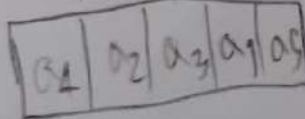
วิธีในหนังสือ

$i = 0 \rightarrow 5$



$j = 0 \rightarrow 5$

do: temp = $a[i]$ temp2 = $a[j]$



เป็นการเก็บตัวน้อยที่สุด
 มาใส่ที่เล็กที่สุด และเก็บไว้ในที่ที่เก็บ

5. ลำดับขั้นการปฏิบัติงาน

5.1. จงเขียนผังงานและโค้ดโปรแกรมเพื่อแก้ไขโจทย์ปัญหาดังต่อไปนี้

5.1.1. จากไฟล์ข้อมูล Salary.txt จงเขียนโปรแกรมเพื่อกระทำการดังต่อไปนี้

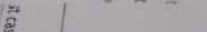
- เรียงลำดับชื่อของคนที่มีอายุมากที่สุดไปยังคนที่มีอายุน้อยที่สุดในบริษัท (เฉพาะ 5 คนแรก)
- เรียงลำดับคนชื่อของคนที่มีเงินเดือนน้อยที่สุด ไปยังคนที่มีเงินเดือนมากที่สุด (เฉพาะ 5 คนแรก)
- กรอกอายุ เพื่อทำการค้นหาชื่อของคนที่มีอายุดังกล่าว

Salary.txt

Name	Age	Salary(Bath)	Job
Kim	25	31,250	Advisor
Pong	32	45,000	Programmer
Som	25	23,000	Support
Aoy	24	33,250	Advisor
Ying	26	34,550	Programmer
DJ	29	21,150	HR
Yot	28	50,000	Founder
Pot	31	24,500	HR
Vip	25	25,450	Programmer
A	21	22,250	Support

Test case 1

Input	Find Age : 29				
Output	32	31	29	28	26
Age (99 - 0) :	Pong	Pot	DJ	Yot	Ying
	21150	22250	23000	24500	25450
Salary(99999 - 0) :	DJ	A	Som	Pot	Vip
Age 29 =	DJ				



5.1.3 จงเขียนโค้ดโปรแกรม

เพื่อโปรแกรม

Lab16

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
```

```
struct employee {
    char name[100];
    int age;
    float salary;
    char job[100];
}typedef worker;
```

```
int collector (worker [] );
void sortAge( worker [], int );
void sortSalary( worker [], int );
void search( worker [], int, int );
```

```
int main() {
    int countL;
    int searcher;
    worker employee[100];
    countL = collector(employee);
    printf("-----\n");
    printf("Find Age : ");
    scanf( "%d", &searcher );
    sortAge( employee, countL );
    sortSalary( employee, countL );
    search( employee, countL, searcher );
}
```

```
int collector (worker take[] ) {
```



```

int count = 0 ;
FILE *fp ;
fp = fopen( "Employee.txt", "r" );
if (fp == NULL ){
    printf( "ERROR : NOT FOUND DATA FILE!" );
    exit( 0 );
}
fscanf( fp, "%s\t%s\t%s\t%s\n", take[0].name, take[0].name, take[0].name, take[0].name );

for (int i = 0 ; i < 100 ; i++){

    if (fscanf( fp, "%s\t%d\t%f\t%s\n", take[i].name, &take[i].age, &take[i].salary,
take[i].job ) != EOF ) {
        count++ ;
        printf("%s\t%d\t%.2f\t%s\n", take[i].name, take[i].age, take[i].salary,
take[i].job) ;
    }
    else{
        break ;
    }
}
fclose(fp) ;
return count ;
}

```

```

void sortAge( worker data[], int num_worker ) {
    worker holdTposi ;
    for (int i = 1; i < num_worker ; i++) {
        for (int j = 0 ; j < i ; j++) {
            if( data[i].age > data[j].age){
                holdTposi = data[i] ;
                for( int k = i ; k > j ; k--){

```



```

        data[k] = data[k-1];
    }
    data[j] = holdTposi;
    break;
}

}

printf("OUTPUT:\n\t\t");
for (int c = 0; c < 5; c++){
    printf("%d\t", data[c].age);
}
printf("\n");
printf("Age( 99 - 0 )\t");
for (int l = 0; l < 5; l++){
    printf("%s\t", data[l].name);
}
printf("\n");
}

```

```

void sortSalary( worker data[], int num_worker ) {
    worker holdTposi;
    for (int i = 1; i < num_worker; i++) {
        for (int j = 0; j < i; j++) {
            if( data[i].salary < data[j].salary){
                holdTposi = data[i];
                for( int k = i; k > j; k--){
                    data[k] = data[k-1];
                }
                data[j] = holdTposi;
            }
            break;
        }
    }
}

```

```

    }
}

printf("OUTPUT:\n\t\t\t");
for (int c = 0 ; c < 5 ; c++){
    printf("%.f\t", data[c].salary);
}
printf("\n");
printf("Salary( 99999 - 0 )\t");
for (int l = 0 ; l < 5 ; l++){
    printf("%s\t", data[l].name);
}
printf("\n");
}

```

```

void search( worker data[], int num_worker, int founder ) {
    printf( "\nAge %d = ", founder );
    for ( int i = 0 ; i < num_worker ; i++ ) {
        if( data[i].age == founder ) {
            printf( "%s\t", data[i].name );
        }
    }
}
}

```

6. สรุปผลการปฏิบัติงาน
 1. การปรับโครงสร้างข้อมูล Solong try มีข้อดีคือใช้
 structure ในทางโครงสร้างและรวมข้อมูล file แล้วมีขนาดคนที่เกี่ยวข้อง
 กับโครงสร้าง การแก้ไขทำได้ง่ายกว่าโครงสร้างแบบ
 2. การแก้ไขข้อมูล โครงสร้างแบบแก้ไขข้อมูลได้โดยตรง

7. คำถามทางการทดลอง

7.1. จงระบุข้อควรระวังในการค้นหาข้อมูล
 ควรใช้การเปรียบเทียบให้ตรงกันกับข้อมูลที่ต้องการ
 เพื่อความถูกต้องในการเขียน

7.2. จงระบุข้อควรระวังในการเรียงข้อมูล
 ควรเรียงลำดับการแทนที่ของค่าที่ต้องการย้ายให้เคลื่อน
 ไปที่หน้าหรือด้านหลัง (เข้า 4 หน้า)

7.3. จงบอกแนวทางการ Swap ข้อมูล
 1. การ swap คือการเปลี่ยนตำแหน่งที่ตั้งของข้อมูล
 2. การ swap ในการเก็บค่าของตัวแปรและค่าของตัวแปรอื่น
 3. การใช้ pointer เพื่อจะทำการเก็บค่าของตัวแปรและค่าของตัวแปรอื่น

7.4. จงระบุความเหมือน/แตกต่างในการเรียงข้อมูลระหว่างตัวแปรธรรมดาและตัวแปรโครงสร้าง
 ยกตัวอย่างละเอียด

การเรียง structure สามารถทำได้ทั้งการเรียงข้อมูล
 แต่การเรียง array จะเรียงข้อมูลได้ทั้งค่าและตำแหน่งที่เก็บ
 ดังนั้น structure จึงเหมาะสำหรับการเก็บข้อมูลที่มีลักษณะ
 1. 1- 1000 การเรียง