

ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

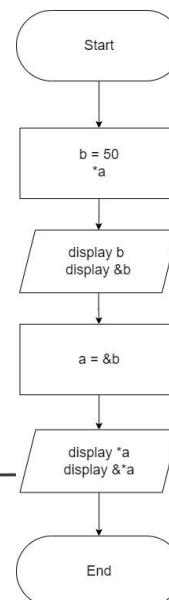
คำอธิบาย	Code ตัวอย่าง
<p>pointer คือตัวแปรที่เก็บค่าที่อยู่ของข้อมูล</p> <p>ของตัวแปร จึงสามารถใช้เข้าถึงข้อมูลได้โดยตรง</p> <p>โดยใช้ชื่อของตัวแปรที่เก็บค่าที่อยู่ของตัวแปร</p>	<pre> 1 #include <stdio.h> 2 //Pointer 3 int main() 4 { 5 int b = 50 ; 6 int *a ; 7 8 printf(" b = %d, &b = %d \n", b, &b); 9 printf("-----\n"); 10 11 a = &b ; 12 printf(" a = %d, &a = %d \n", *a, &a); 13 14 return 0; 15 }</pre>

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

Flow chart ของ Code ตัวอย่าง

```

C:\Users\armwo\Desktop\icon  X  +  -
b = 50, &b = 6487572
-----
a = 50, &a = 6487572
-----
Process exited after 0.02968 seconds with return value 0
Press any key to continue . . .
```



ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

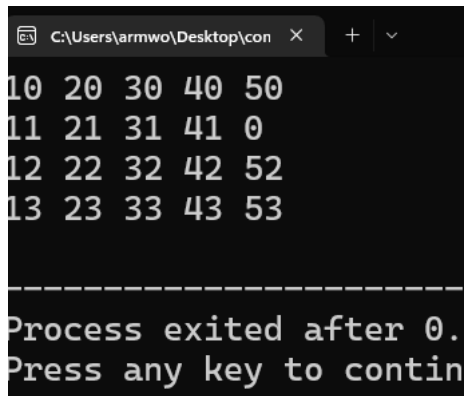
คำอธิบาย

กำหนด pointer Array 1 ตัว และทำการเก็บค่าในแถวของ Array แถว 1 ถึงแถว 4 ของ Array ที่ 51091 จากนั้นทำการเปลี่ยนค่าใน pointer ด้วย if, for loop ทำการแสดงผลของ Array ออกมาโดยการ printf

Code ตัวอย่าง

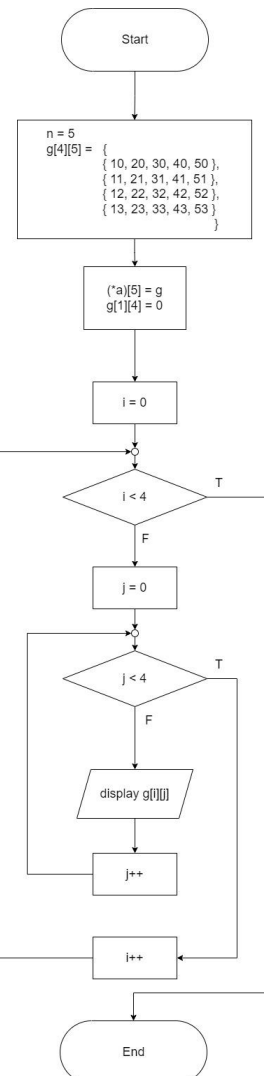
```
1 #include <stdio.h>
2
3 int main() {
4     int n = 5 ;
5     int g[4][5] = {
6         { 10, 20, 30, 40, 50 },
7         { 11, 21, 31, 41, 51 },
8         { 12, 22, 32, 42, 52 },
9         { 13, 23, 33, 43, 53 }
10    };
11
12    int (*a)[5] = g ;
13    g[1][4] = 0 ;
14
15    for( int i = 0 ; i < 4 ; i++ ) {
16        for( int j = 0 ; j < 5 ; j++ ){
17            printf("%d ", g[i][j] );
18        }
19        printf("\n");
20    }
21    return 0;
22 }
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)



```
10 20 30 40 50
11 21 31 41 0
12 22 32 42 52
13 23 33 43 53

Process exited after 0.
Press any key to continue
```



ข้อที่ 3 จงอธิบายเรื่อง Pointer Function/ตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

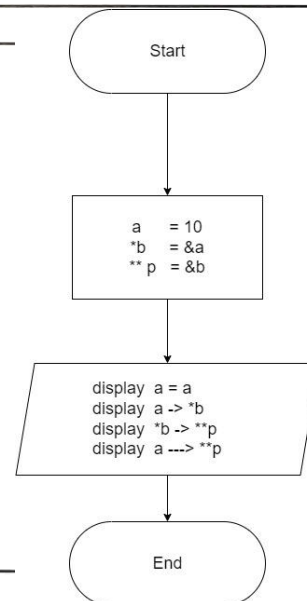
เป็นฟังก์ชันที่ส่งค่าของตัวแปรไปเป็นค่าให้กับฟังก์ชันอื่น
หรือ pointer ซึ่งใช้เพื่อเก็บค่าของตัวแปร
สามารถมองเป็นจุดๆ และค่าของตัวแปร
ที่เก็บไว้

Code ตัวอย่าง

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5     int a = 10 ;
6     int *b = &a ;
7     int **p = &b ;
8
9     printf("a = %d\n-----\n", a);
10
11     printf("a -> *b = %d\n\n", *b);
12
13     printf("*b -> **p = %d\n\n", **p);
14
15     printf("a ----> **p = %d\n\n", **p);
16
17     return 0;
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

```
C:\Users\armwo\Desktop\con x + v
a = 10
-----
a -> *b = 10
*b -> **p = 10
a ----> **p = 10
-----
Process exited after 0.009681 seconds with return value 0
Press any key to continue . . .
```



ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

Dynamic Array เช่น แรร์อาร์เรย์
ที่ใช้เก็บข้อมูลแบบไดนามิกที่ไม่รู้
ขนาดที่แน่นอน เพื่อปรับขนาดอาร์เรย์
จากรุ่นก่อนหน้า จะปรับขนาดให้เหมาะสม

Code ตัวอย่าง

```
1 #include <stdio.h>
2 int main()
3 {
4     int row, col ;
5     printf("How many rows , cols ? \n");
6     scanf("%d", &row);
7     scanf("%d", &col);
8
9     for(int i = 0 ; i < row ; i++){
10         for(int j = 0 ; j < col ; j++){
11             printf("\narr[%d][%d] : ", i, j);
12         }
13     }
14     return 0;
15 }
```

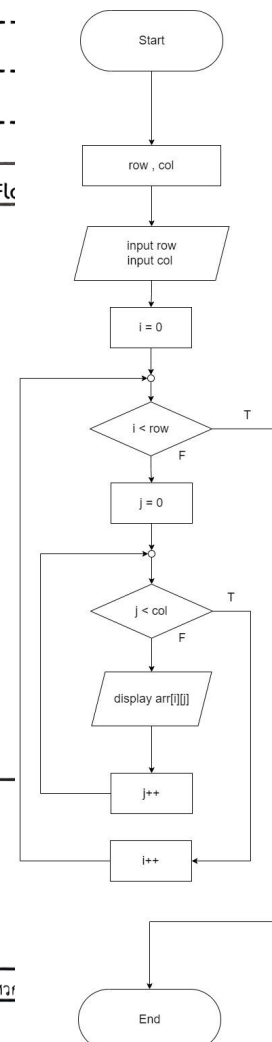
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)



```
C:\Users\armwo\Desktop\con X + v
How many rows , cols ?
5
5

arr[0][0] :
arr[0][1] :
arr[0][2] :
arr[0][3] :
arr[0][4] :
arr[1][0] :
arr[1][1] :
arr[1][2] :
arr[1][3] :
arr[1][4] :
arr[2][0] :
arr[2][1] :
arr[2][2] :
arr[2][3] :
arr[2][4] :
arr[3][0] :
arr[3][1] :
arr[3][2] :
arr[3][3] :
arr[3][4] :
arr[4][0] :
arr[4][1] :
arr[4][2] :
arr[4][3] :
arr[4][4] :
-----
```

Flowchart



ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

Code ตัวอย่าง

การส่งผ่านตัวแปร คือการส่งค่าตัวแปร
เช่น อาร์เรย์ของตัวเลข ฟังก์ชัน ฟังก์ชัน
ตัวแปรที่อยู่ใน function จะไม่ผล
ให้กับตัวแปรที่อยู่นอก function ด้วย.

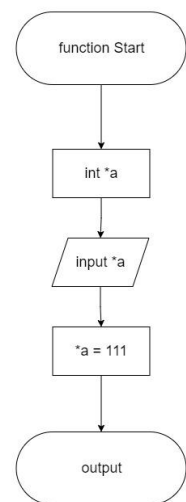
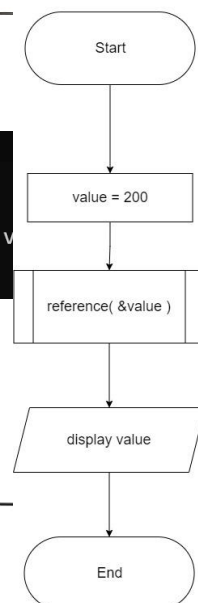
```
1 #include<stdio.h>
2
3 int reference(int *a){
4     *a = 111;
5 }
6
7 int main() {
8     int value = 200;
9     reference(&value);
10
11     printf("value = %d",value);
12 }
```

ผลลัพธ์ของ Code (Captureพร้อมแปรรูป)

```
C:\Users\jarmwo\Desktop\con x + v
value = 111
-----
Process exited after 0.01214 seconds with return v
Press any key to continue . . . |
```

main function

Reference function



ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

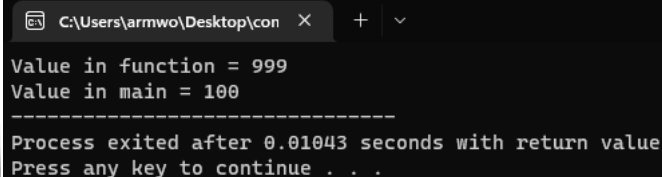
คำอธิบาย

การส่งค่า (value) เข้าออกผ่านตัวแปร
ของฟังก์ชัน จะไม่ส่งผลกระทบต่อตัวแปรนอก
ฟังก์ชัน

Code ตัวอย่าง

```
1 #include <stdio.h>
2
3 int pbvalue( int var ){
4     var = 999 ;
5
6     printf("Value in function = %d \n", var);
7 }
8
9 int main(){
10
11     int var = 100 ;
12     pbvalue( var );
13
14     printf("Value in main = %d", var);
15     return 0;
16 }
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)



```
C:\Users\armwo\Desktop\con
Value in function = 999
Value in main = 100
-----
Process exited after 0.01043 seconds with return value 0
Press any key to continue . . .
```

