

ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

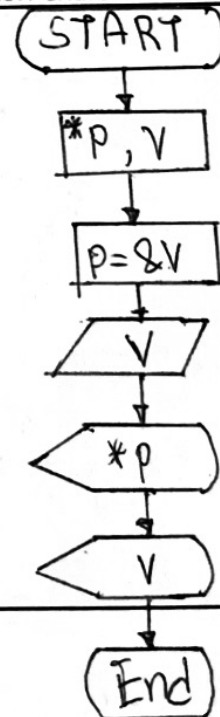
คำอธิบาย	Code ตัวอย่าง
<p>Pointer คือตัวชี้ จะเก็บ ค่า ที่อยู่ ของ หน่วย ความจำ ซึ่งใช้ประโยชน์ในการจัดการข้อมูล Dynamic data structure ซึ่งรูปแบบ ชนิดข้อมูล * ชื่อตัวแปร เช่น <code>int *variable;</code> โดยตัวแปรเก็บค่าตัวแปร และชนิดข้อมูลเหมือนกัน กัน ซึ่งสุดท้ายตัว pointer ก็จะจำวนารถชี้ไป ที่ ตัวแปรนั้นๆ จะใช้เครื่องหมาย & ชี้ไป ที่ address ของตัวแปร นั้น เมื่อตัวแปร pointer ชี้ไปอยู่ที่ตัวแปรนั้นๆ เมื่อตัวแปรชี้ค่า เปลี่ยนไป ตัวแปร pointer จะเปลี่ยนตามไปด้วย</p>	<pre>#include <stdio.h> int main () { int *p; int v; p = &v; printf("Enter your value : "); scanf("%d", &v); printf("----OUTPUT----\n"); printf("*p=%d\n", *p); printf("v=%d", v); return 0; }</pre>

ผลลัพธ์ของ Code (Captureพร้อมแปรรูป)

```
Enter your value : v
----OUTPUT----
*p = 0
v = 0

Process exited after 13.63 seconds with return value 0
Press any key to continue . . .
```

Flow chart ของ Code ตัวอย่าง



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

คำอธิบาย

Code ตัวอย่าง

โดยจะสร้าง array 2 มิติแบบปกติ
และกำหนดค่าไป แล้ว for loop
ย้อนกลับในการ printf แถวละ 10 คอลัมน์
ในกำหนดแถว 5 แถว (*n)[5] = c;
โดยค่า c จะไปเก็บที่ (*n) 5 แถว
และสองแถวใน array 2 มิติ
for loop printf ค่าออกมาเหมือนเดิม
โดยจะให้ c[a][b] แทนเหมือนเดิม เพราะ
กำหนดค่าไป ตัวแปรที่ ชื่อ ก็เปลี่ยนตาม

```
#include <stdio.h>

int main () {
    int a, b ;
    int c[4][5] = {
        { 10, 20, 30, 40, 50 },
        { 11, 21, 31, 41, 51 },
        { 12, 22, 32, 42, 52 },
        { 13, 23, 33, 43, 53 }
    };

    for ( a = 0 ; a < 4 ; a++ ) {
        for( b = 0 ; b < 5 ; b++ )
            printf( "%d ", c[a][b] );
        printf( "\n" );
    }

    printf( "\n-----\n" );

    int (*n)[5] = c ;

    n[3][3] = 89 ;
    n[3][0] = 70 ;

    for ( a = 0 ; a < 4 ; a++ ) {
        for( b = 0 ; b < 5 ; b++ )
            printf( "%d ", c[a][b] );
        printf( "\n" );
    }

    return 0 ;
}
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

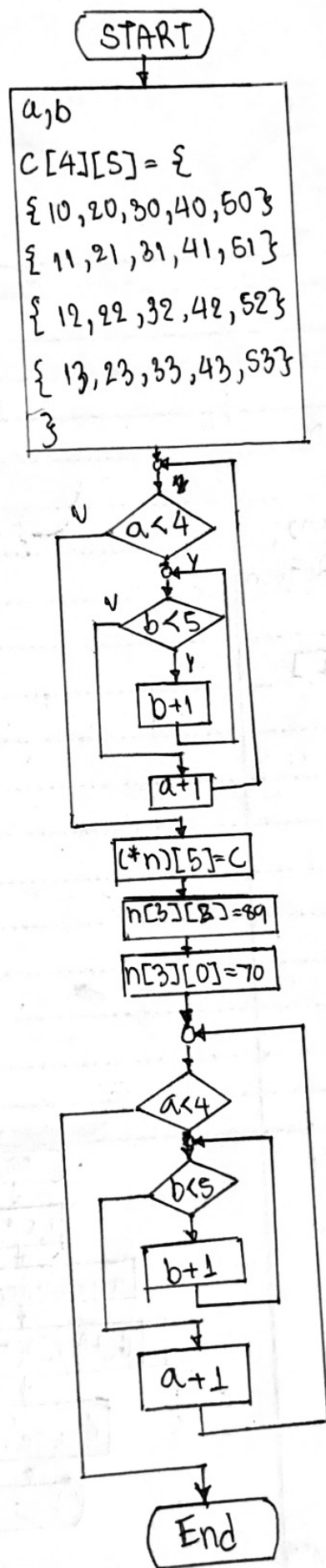
Flow chart ของ Code ตัวอย่าง

```
10 20 30 40 50
11 21 31 41 51
12 22 32 42 52
13 23 33 43 53

10 20 30 40 50 11 21 31 41 51 12 22 32 42 52 70 23 33 89 53

Process exited after 0.01283 seconds with return value 0
Press any key to continue . . .
```

Flowchart 402



ข้อที่ 3 จงอธิบายเรื่อง Pointer Function ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

เริ่มแรกเราต้องดูโค้ดจาก main ก่อนแล้วค่อยไปดูฟังก์ชันย่อย Pointer ไม่ได้ชี้ตัวแปรโดยตรงเลย ยังสามารถส่งค่าส่งได้ ขาดตัวการประกาศ argument ยังไม่ต้องใส่ชื่อตัวแปรเข้าไปได้ และสามารถนำค่าผลลัพธ์ใช้กับโค้ดที่ทำงานได้

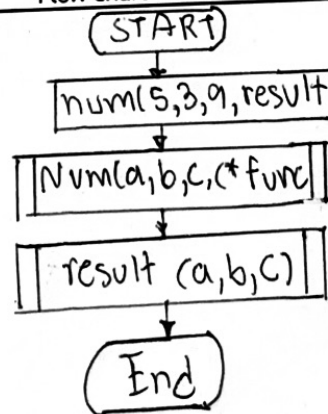
Code ตัวอย่าง

```
#include <stdio.h>
int result(int a, int b, int c) {
return
    return a + (b * c);
}
int num(int a, int b, int c, int (*func)(int, int, int)) {
    return (*func)(a, b, c);
}
int main () {
    printf("OUTPUT=%d", num(
        5, 3, 9, result));
    return 0;
}
```

ผลลัพธ์ของ Code (Capture พร้อมแปรรูป)

OUTPUT = 5
Process exited after 0.03309 seconds with return value 0
Press any key to continue . . .

Flow chart ของ Code ตัวอย่าง



ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

Code ตัวอย่าง

* 6 จัดสรรหน่วยที่เก็บ array ได้เหมือนกัน
ถ้ายังไม่จัดสรร array ได้ใช้
Dynamic array จัดสรรได้ไม่จำกัด
ใช้เท่าไรก็ได้ จะประกาศค่าลงหน้าของก็ได้
ต่อไปทำ array ที่ใช้สมาชิกที่ตัวก็ได้
โดยที่เราไปจองหน่วยความจำ
int *b; 0 1 2...n 10b4
b = new int [n]; b [] ของ Address
ของ 2 มิติ ทำได้ 2 อย่าง
1) int (*c)[4] int **c; 2) int c[4][4]
C = new int [n][4]
ทั้ง array 2 มิติ int a[3][4] ต้องขาด 12 หน่วย
Array 3x4 0 1 2 3 4 5 6 7 8 9 10 11
a[6] = a [1x4 + 2] a[i x col + j]

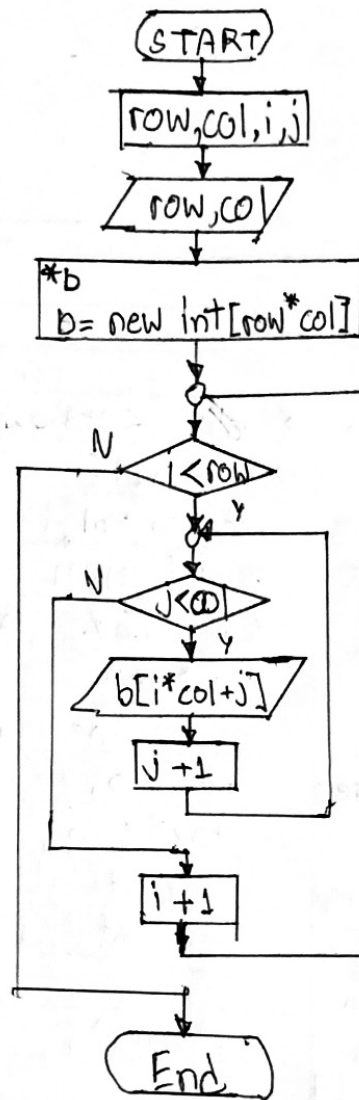
```
#include <stdio.h>
int main() {
    int row, col, i, j;
    printf("INPUT : ");
    scanf("%d %d", &row, &col);
    int *b;
    b = new int [row * col];
    for (i = 0; i < row; i++)
        for (j = 0; j < col; j++) {
            printf("a[%d][%d] : ", i, j);
            scanf("%d", &b[i * col + j]);
        }
    return 0;
}
```

ผลลัพธ์ของ Code (Captureพร้อมแป้นพิมพ์)

Flow chart ของ Code ตัวอย่าง

```
INPUT : 3 3
a[0][0] : 10
a[0][1] : 20
a[0][2] : 30
a[1][0] : 11
a[1][1] : 21
a[1][2] : 31
a[2][0] : 12
a[2][1] : 22
a[2][2] : 32
.....
Process exited after 29.23 seconds with return value 0
Press any key to continue . . .
```

Flow Chart 904

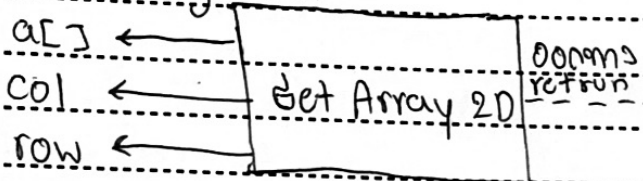


ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

Code ตัวอย่าง

Pass by reference โดยที่เมื่อส่งค่าของ Argument และ Argument ที่ถูกส่งเข้ามาในตัวแปรใน array ก็เป็น pointer ดังนั้น array ที่ถูกส่งเข้ามาใน Pointer จะได้รับค่าที่เก็บอยู่ 1 ระดับของ Argument

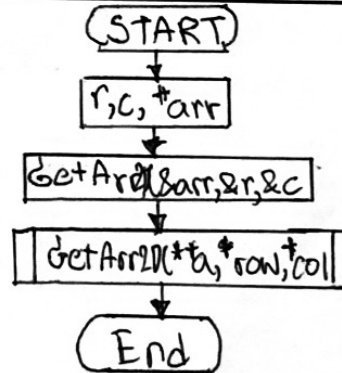


```
--#include <stdio.h>
--
--void GetArr2D( int **a, int *row, int *col );
--
--int main() {
--    int r, c, *arr ;
--    GetArr2D( &arr, &r, &c );
--    return 0 ;
--}
--
--void GetArr2D( int **a, int *row, int *col ) {
--    int i, j;
--    printf( "How many row, col ? " );
--    scanf( "%d %d", row, col );
--    *a = new int[ *row * *col ];
--    for( i = 0 ; i < *row ; i++ )
--        for( j = 0 ; j < *col ; j++ ) {
--            printf( "a[%d][%d] = ", i, j );
--            scanf( "%d", &(*a)[ i * *col + j ] );
--        }
--}
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

Flow chart ของ Code ตัวอย่าง

```
How many row, col ? 2 2
a[0][0] = 00
a[0][1] = 01
a[1][0] = 10
a[1][1] = 11
```



ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย

Pass by Value คือ การส่งค่า
ค่าไปให้ ทั้ง 2 ทว Pass by Value
1 ทว และ by Reference 2 ทว
คือ Argument คือ Return
col ← Get Array 2D → a[]
row ←

Code ตัวอย่าง

```
#include <stdio.h>

int *GetArr2D( int *row, int *col );

int main() {
    int r, c, *arr;
    arr = GetArr2D( &r, &c );
    delete [] arr;
    return 0;
}

int *GetArr2D( int *row, int *col ) {
    int *a;
    int i, j;
    printf( "How many row, col ? " );
    scanf( "%d %d", row, col );
    a = new int[ *row * *col ];
    for( i = 0 ; i < *row ; i++ )
        for( j = 0 ; j < *col ; j++ ) {
            printf( "a[%d][%d] = ", i, j );
            scanf( "%d", &a[ i * *col + j ] );
        }
    return a;
}
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

```
How many row, col ? 4 4
a[0][0] = 0000
a[0][1] = 0001
a[0][2] = 0010
a[0][3] = 0011
a[1][0] = 0100
a[1][1] = 0101
a[1][2] = 0110
a[1][3] = 0111
a[2][0] = 1000
a[2][1] = 1001
a[2][2] = 1010
a[2][3] = 1011
a[3][0] = 1100
a[3][1] = 1101
a[3][2] = 1110
a[3][3] = 1111
```

Flow chart ของ Code ตัวอย่าง

