

## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวิเคราะห์ประกอบตามความเข้าใจของคุณ

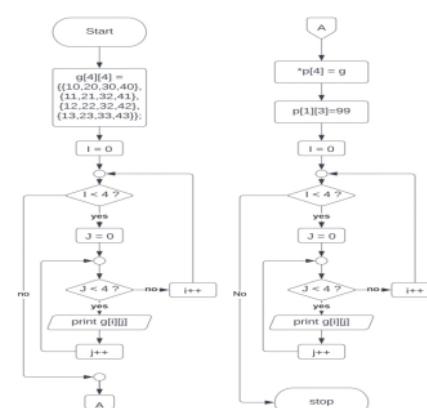
ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

คำอธิบาย	Code ตัวอย่าง
<p>1. คือ Pointer ในการสื่อสารว่าต้องการดึงหน้าไปที่ต้อง หันกลับด้านกัน ก็ต้อง</p> <p>เมื่อเราทำแบบนี้ Pointer ให้เราเก็บงานหน้างาน * ไว้หน้า หน้า ณ ตำแหน่งที่เราหันหน้า ให้เป็น ตัว Pointer</p>	<pre> 1 #include &lt;stdio.h&gt; 2 int main() { 3     int *a; 4     int var = 15; 5     a = &amp;var; 6     printf("*a = %d\n", *a); 7     var = 25; 8     printf("*a = %d\n", *a); 9     *a = 35; 10    printf("*a = %d\n", *a); 11    printf("var = %d\n", var); 12    return 0; 13 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD     Start((Start)) --&gt; Init["*a, var = 0"]     Init --&gt; Assign["a = &amp;var"]     Assign --&gt; Out1[OUTPUT *a]     Out1 --&gt; Var25["var = 25"]     Var25 --&gt; Out2[OUTPUT *a]     Out2 --&gt; A35["a = 35"]     A35 --&gt; Out3[OUTPUT *a]     Out3 --&gt; Var35["var = 35"]     Var35 --&gt; OutputVar[OUTPUT var]     OutputVar --&gt; Stop((STOP)) </pre>

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

คำอธิบาย	Code ตัวอย่าง
<p>Pointer คืออะไร Array คือ รูปแบบ Array ที่มีไวด์กู้ดแบบ กล่าวคือ แต่ละแถวจะต้องมีตัวแปรต่อไปนี้คงส่วนที่ทุกครั้ง ที่เปลี่ยนต้องมี Pointer ในการระบุหนึ่งใน Array กัน</p>	<pre> 1 #include &lt;stdio.h&gt; 2 int main() { 3     int g[4][4] = {{10, 20, 30, 40}, 4                    {11, 21, 31, 41}, 5                    {12, 22, 32, 42}, 6                    {13, 23, 33, 43}}; 7 8     for (int i = 0; i &lt; 4; i++) { 9         for (int j = 0; j &lt; 4; j++) { 10            printf("%d ", g[i][j]); 11        } 12        printf("\n"); 13    } 14    printf("-----\n"); 15    int(*p)[4] = g; 16    p[1][3] = 99; 17    for (int i = 0; i &lt; 4; i++) { 18        for (int j = 0; j &lt; 4; j++) { 19            printf("%d ", g[i][j]); 20        } 21        printf("\n"); 22    } 23    return 0; 24 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง

PS D:\NWE\_NUT\С\LAB01>  
10 20 30 40  
11 21 31 41  
12 22 32 42  
13 23 33 43  
-----  
10 20 30 40  
11 21 31 99  
12 22 32 42  
13 23 33 43



## ข้อที่ 3 จงอธิบายเรื่อง Pointer Function ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pointer function คือ function ที่รับ function หรือฟังก์ชัน function ทางเข้ามาเป็น parameter ของตัวเอง ทำให้สามารถเรียกใช้ function ทางเดียวได้โดยไม่ต้องระบุชื่อ function แต่จะระบุชื่อฟังก์ชันที่ต้องการใช้ เช่น <code>_operation</code> คือฟังก์ชันที่รับ <code>x</code> และ <code>y</code> แล้วคืนค่า <code>x + y</code> กลับไป</p>	<pre> 1 #include &lt;stdio.h&gt; 2 int _add(int x, int y) { 3     return x + y; 4 } 5 int _operation(int x, int y, int (*function)(int, int)){ 6     return (*function)(x, y); 7 } 8 int main(){ 9     printf("==&gt; %d\n", _operation(26, 8, _add)); 10    return 0; 11 }</pre>
ผลลัพธ์ของ Code (Capture พร้อมภาพรูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD     start([start]) --&gt; operation1[_operation]     operation1 --&gt; add1[_add]     add1 --&gt; result1[x + y]     result1 --&gt; return1[return]     return1 --&gt; print[print ==&gt; _add]     print --&gt; end([end])     </pre>

## ข้อที่ 4 จะอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p><u>Dynamic Array คือ Array ที่สามารถเพิ่มหรือลดจำนวนได้</u>  <u>มันต้อง Array ที่ไม่รู้ว่ามีกี่ช่องในหนึ่งเดียว แต่ต้องรู้ว่ามีกี่ช่องในทั้งหมด</u>  <u>ก็ pointer ที่มีค่า Array ไม่ต้องรู้ว่ามีกี่ช่อง Dynamic Array</u></p>	<pre> 1 #include &lt;stdio.h&gt; 2 int main() { 3     int row, col, i, j,*a; 4     printf("how many rows and col ? : "); 5     scanf("%d %d", &amp;row, &amp;col); 6     a = new int[row * col]; 7     for (int i = 0; i &lt; row; i++) { 8         for (int j = 0; j &lt; col; j++) { 9             printf("a[%d][%d]: ", i, j); 10            scanf("%d", &amp;a[i * col + j]); 11        }//end Loop j 12    } // end Loop i 13    printf("\n"); 14    return 0; 15 }</pre>
ผลลัพธ์ของ Code (Capture พร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย  
Pass by reference คือ การส่งตัวแปรไปใน function นั้นๆ ไม่ได้ส่ง value ไปใน function นั้นๆ แต่ส่ง address ไปใน function นั้นๆ แล้ว function นั้นๆ จะเปลี่ยนแปลง value ที่อยู่ใน memory ของตัวแปรที่ส่งไป

Code ตัวอย่าง

```

1 #include <stdio.h>
2 void GetArr2D(int **a, int *row, int *col) {
3     printf("How many rows and columns ? : ");
4     scanf("%d %d", row, col);
5     *a = new int[*row * *col];
6     for (int i = 0; i < *row; i++) {
7         for (int j = 0; j < *col; j++) {
8             printf("a[%d][%d]: ", i, j);
9             scanf("%d", &(*a)[i * *col + j]);
10        } // end loop j
11    } // end loop i
12 } // end Getarr2d
13 void GetArr2D(int **a, int *row, int *col); // prototype
14 int main() {
15     int r, c, *arr;
16     GetArr2D(&arr, &r, &c);
17     return 0;
18 }
```

ผลลัพธ์ของ Code (Capture พร้อมแปะรูป)

```

PS D:\NWE_NUT\C> cd "d:\NWE_NUT\C\LAB01\"  

How many rows and columns ? : 3 4  

a[0][0]: 11  

a[0][1]: 12  

a[0][2]: 13  

a[0][3]: 14  

a[1][0]: 21  

a[1][1]: 22  

a[1][2]: 23  

a[1][3]: 24  

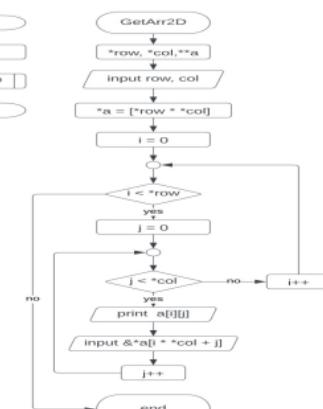
a[2][0]: 31  

a[2][1]: 32  

a[2][2]: 33  

a[2][3]: 34
```

Flow chart ของ Code ตัวอย่าง



## ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by value คือการส่งค่าที่ทางโปรแกรมจะนำ回去 Return      ใน function main จะเรียก function สอง ต่อไปนี้ function สอง      นี้ค่าที่เก็บไว้ใน function main ไม่ได้ Return.</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 int *GetArr2D(int *row, int *col) { 4     printf("How many row and col ? : "); 5     scanf("%d %d", row, col); 6     int *arr; 7     arr = new int[*row * *col]; 8     for (int i = 0; i &lt; *row; i++) { 9         for (int j = 0; j &lt; *col; j++) { 10            printf("a[%d][%d] = ", i, j); 11            scanf("%d", &amp;arr[i * *col + j]); 12        } // end Loop j 13    } // end Loop i 14    return arr; 15 } // end GetArr2D 16 17 int *GetArr2D(int *row, int *col); // prototype 18 19 int main() { 20     int r, c, *arr; 21     arr = GetArr2D(&amp;r, &amp;c); 22     delete[] arr; 23     return 0; 24 }</pre>

ผลลัพธ์ของ Code (Capture พร้อมภาพรูป)	Flow chart ของ Code ตัวอย่าง
<pre> PS D:\NWE_NUT\C\LAB01&gt; cd "d:\NWE_NUT\C\LAB01" How many row and col ? : 2 2 a[0][0]: = 34 a[0][1]: = 45 a[1][0]: = 67 a[1][1]: = 87 PS D:\NWE_NUT\C\LAB01&gt;</pre>	<pre> graph TD     start([start]) --&gt; r_c_arr[r, c, arr]     r_c_arr --&gt; GetArr2D[GetArr2D]     GetArr2D --&gt; InputRowCol[/Input row, col/]     InputRowCol --&gt; arrInit[arr = (*row * *col)]     arrInit --&gt; i0[i = 0]     i0 --&gt; iLessRow{i &lt; *row}     iLessRow -- yes --&gt; j0[j = 0]     j0 --&gt; jLessCol{j &lt; *col}     jLessCol -- yes --&gt; printA[a[i][j]]     printA --&gt; inputArr[/Input &amp;a[0] * *col + i/]     inputArr --&gt; jPlus1[j++]     jLessCol -- no --&gt; iPlus1[i++]     iLessRow -- no --&gt; returnArr{return arr}     </pre>