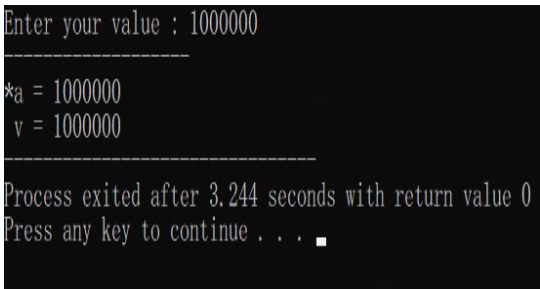
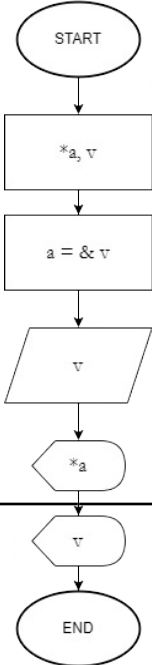


## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>Pointer คือ ตัวชี้ ที่ถือถือเป็นลักษณะเด่น Operator ในภาษา C สำหรับใช้แทนตัวแปรข้อมูล</p> <p>แบบ Dynamic data structure</p> <p>ตัวแปรชนิด Pointer จะเก็บค่า "ที่อยู่" ของหน่วยความจำ ( แตกต่าง กับตัวแปรชนิดอื่นที่เก็บแค่ค่าจริง )</p>	<pre>#include &lt;stdio.h&gt;  int main () {     int *a ;     int v ;      a = &amp;v ;     printf("Enter your value : ");     scanf ("%d", &amp;v );     printf ("-----\n");     printf ("*a = %d /n", *a );     printf ("v = %d /n", v );     return 0 ; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre>graph TD     START([START]) --&gt; Init[*a, v]     Init --&gt; Assign[a = &amp;v]     Assign --&gt; Input[/v/]     Input --&gt; PrintA{{*a}}     PrintA --&gt; PrintV{{v}}     PrintV --&gt; END([END])</pre>

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

คำอธิบาย

โดยคำสั่ง array 2 มิติแบบปกติ เราจะกำหนดค่าลงไปและใช้ for loop ใส่อะไรใน printf แถวคอลัมน์ใน array 2 มิติ และ ประยุกต์ใช้ array ไม่จำกัดแถวสร้างแบบ (\*n)[5] = c; โดยค่า c จะเก็บที่ (\*n) เรียงซ้อน และลองเปลี่ยนค่าใน array 2 มิติใช้ For loop printf ค่าออกมาเหมือนเดิม เพราะกำหนดค่าไว้ ตัวแปรที่ชี้ก็เปลี่ยนตาม

Code ตัวอย่าง

```
1 #include <stdio.h>
2 int main(){
3     int a , b ;
4     int g[4][5]={
5         { 10, 20, 30, 40, 50 },
6         { 11, 21, 31, 41, 51 },
7         { 12, 22, 32, 42, 52 },
8         { 13, 23, 33, 43, 53 }
9     };
10    for (a = 0 ; a < 4 ; a++){
11        for(b=0 ; b < 5 ; b++){
12            printf("%d ", g[a][b]);
13            printf("\n");
14        }
15        printf("-----\n");
16        int (*n)[5] = g;
17        n[3][3] = 69 ;
18        n[2][0] = 96 ;
19
20        for (a = 0 ; a < 4 ; a++){
21            for(b = 0 ; b < 5 ; b++){
22                printf("%d ", g[a][b]);
23                printf("\n");
24            }
25        }
26        return 0;
27    }
```

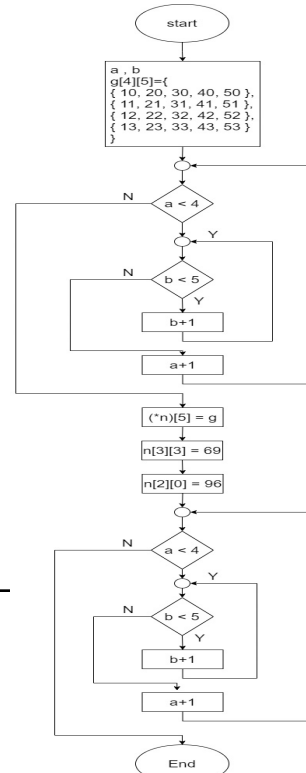
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

Flow chart ของ Code ตัวอย่าง

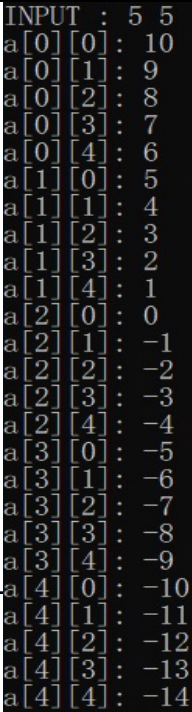
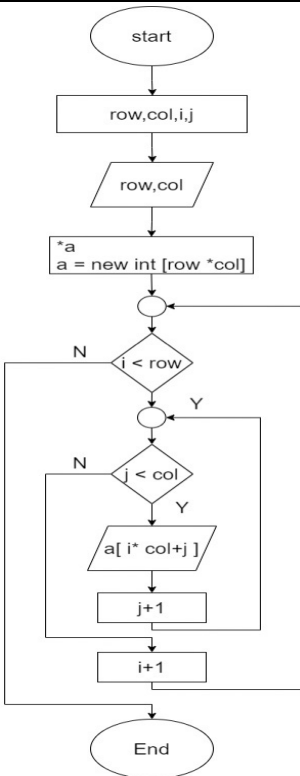
```
10 20 30 40 50
11 21 31 41 51
12 22 32 42 52
13 23 33 43 53
```

```
10 20 30 40 50
11 21 31 41 51
96 22 32 42 52
13 23 33 69 53
```

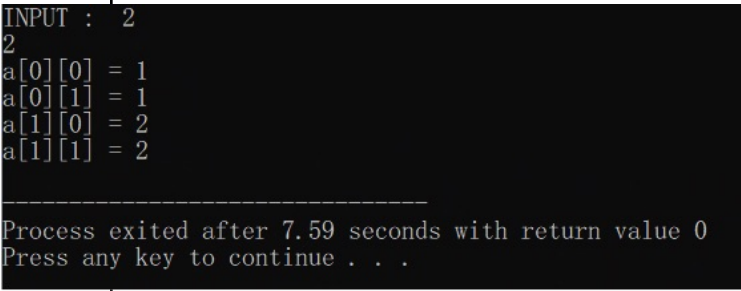
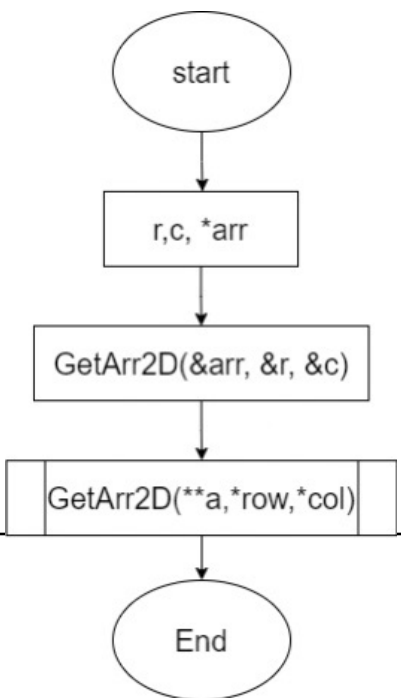
Process exited after 0.02305 seconds with return value 0  
Press any key to continue . . .





ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>..... ใช้ในตัวแปรอาร์เรย์ที่ใช้สำหรับเก็บข้อมูลของตัวแปร ที่คุณ.....          ไม่ทราบจำนวนที่แน่นอน หรือคุณทราบแต่ต้องการ.....          ใช้กับอาร์เรย์แบบไดนามิก เพื่อปรับขนาดของอาร์เรย์ โดยที่ตัวแปร.....          แบบอาร์เรย์จะปรับขนาดจำนวนสมาชิกให้เท่ากับ จำนวนตัวแปร.....          ที่คุณต้องการใช้ในขณะนั้น จะเห็นได้ สดุดีใช้กับอาร์เรย์แบบ ไดนามิก.....          เท่านั้น.....</p>	<pre> #include &lt;stdio.h&gt; int main () {     int row , col , i , j ;     printf ("Input : ");     scanf ("%d %d", &amp;row , &amp;col);     int *a;     a = new int [ row * col ];     for ( i = 0 ; i &lt; row ; i++)         for ( j = 0 ; j &lt; col ; j++) {             printf ("a [%d] [%d] : ", i , j);             scanf ("%d", &amp;a [ i * col + j ]);         }     return 0 ; } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
 <pre> INPUT : 5 5 a[0][0]: 10 a[0][1]: 9 a[0][2]: 8 a[0][3]: 7 a[0][4]: 6 a[1][0]: 5 a[1][1]: 4 a[1][2]: 3 a[1][3]: 2 a[1][4]: 1 a[2][0]: 0 a[2][1]: -1 a[2][2]: -2 a[2][3]: -3 a[2][4]: -4 a[3][0]: -5 a[3][1]: -6 a[3][2]: -7 a[3][3]: -8 a[3][4]: -9 a[4][0]: -10 a[4][1]: -11 a[4][2]: -12 a[4][3]: -13 a[4][4]: -14 </pre>	 <pre> graph TD     Start([start]) --&gt; Init[ row,col,i,j ]     Init --&gt; Input[/row,col/]     Input --&gt; Alloc[ *a a = new int [row * col] ]     Alloc --&gt; Loop1(( ))     Loop1 --&gt; Cond1{ i &lt; row }     Cond1 -- N --&gt; End([End])     Cond1 -- Y --&gt; Loop2(( ))     Loop2 --&gt; Cond2{ j &lt; col }     Cond2 -- N --&gt; IncI[ i+1 ]     IncI --&gt; Loop1     Cond2 -- Y --&gt; Print[/a[ i * col + j ]/]     Print --&gt; IncJ[ j+1 ]     IncJ --&gt; Loop2     IncI --&gt; Loop1     End --&gt; End </pre>

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by reference โดยที่เราส่งค่าของ Argument และ Argument ที่ถูกส่งเข้ามาไว้ที่ตัวแปร array ก็เป็น pointer ดังนั้น array จะออกเป็นเหมือน Pointer ระดับของตัวแปรเอา</p> <p>ตัวอย่างจะตัวแปรหลักค่าเดิมอยู่ 1 ระบุ</p> <p>ส่งค่า Argument</p> <p>arr, col, row</p> <p>Get Array 2D</p> <p>ส่งค่า Return</p>	<pre> 1  #include &lt;stdio.h&gt; 2 3  void GetArr2D(int **a, int *row, int *col); 4 5  int main () { 6      int r, c, *arr; 7      GetArr2D( &amp;arr, &amp;r, &amp;c); 8      return 0 ; 9  } 10 11 void GetArr2D(int **a, int *row, int *col){ 12     int i, j; 13     printf 14     ("INPUT : "); 15     scanf("%d %d", row, col); 16     *a = new int[*row * *col]; 17     for( i = 0 ; i &lt; *row ; i++) 18         for( j = 0 ; j &lt; *col ; j++){ 19             printf("a[%d][%d] = ", i, j); 20             scanf("%d", &amp;(*a)[i* *col + j]); 21         } 22 } 23 </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD     Start([start]) --&gt; Init[r, c, *arr]     Init --&gt; Call1[GetArr2D(&amp;arr, &amp;r, &amp;c)]     Call1 --&gt; Call2[GetArr2D(**a, *row, *col)]     Call2 --&gt; End([End])     </pre>

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by value คือ การเอาค่าออกได้</p> <p>ทั้ง 2 ทฤษฎี Pass by Value 1 ทฤษฎี</p> <p>และ by Reference 2 ทฤษฎี</p> <p>ออกทาง Argument</p> <p>col ←</p> <p>row ←</p> <p>Get Array 2D</p> <p>ออกทาง Return</p> <p>a[]</p>	<pre> 1  #include &lt;stdio.h&gt; 2  int *GetArr2D(int *row, int *col); 3  int main(){ 4      int r, c, *arr; 5      arr = GetArr2D(&amp;r,&amp;c); 6      delete [] arr; 7      return 0; 8  } 9 10 int *GetArr2D(int *row, int *col){ 11     int *a; 12     int i, j; 13     printf("INPUT : " ); 14     scanf("%d %d", row,col); 15     a = new int[*row * *col]; 16     for (i = 0 ; i &lt; *row ; i++){ 17         for( j = 0 ; j &lt; *col ; j++){ 18             printf("a[%d][%d] = ", i ,j); 19             scanf("%d", &amp;a[i * *col + j]); 20         } 21         return a; 22     } </pre>

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD     Start([start]) --&gt; Init[r,c, *arr]     Init --&gt; Call[arr = GetArr2D(&amp;arr, &amp;r, &amp;c)]     Call --&gt; Call2[*GetArr2D(*row,*col)]     Call2 --&gt; Delete[delete [] arr]     Delete --&gt; End([End]) </pre>