

ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จะเขียนคำอธิบาย ยกตัวอย่างประกอบ และวิเคราะห์ประกอบตามความเข้าใจของคุณ

ข้อที่ 1 ช่วยอธิบายความหมายของ Pointer ยังไงและเขียน แม่นยำพื้นที่ที่ใช้ในการเขียน พร้อม Code ผังอธิบายการใช้งาน Pointer

คำอธิบาย	Code ตัวอย่าง
<p>- pointer กือตัวที่ รึ่งกือไปยังจุดที่ไม่แน่นอน Operator ในภาษา C จึงเป็นเครื่องในการจัดการบัญชีมูลค่าแบบ Dynamic data structure ตัวแป้นนั้นคือ Pointer จะเก็บค่า "ที่อยู่" ของหน่วยความจำใน แต่ละตำแหน่ง ตัวแป้นนั้นคือตัวที่เก็บค่าไว้</p>	<pre> ● ● ● 1 #include<stdio.h> 2 3 int main() 4 { 5 int *a ; 6 int var = 10 ; 7 a = &var ; 8 printf("*a = %d\n", *a); 9 10 var = 20 ; 11 printf("*a = %d\n", *a); 12 13 *a = 30 ; 14 printf("*a = %d\n", *a); 15 printf("var = %d\n", var); 16 } 17 </pre>

ผลลัพธ์ของ Code (Captureพร้อมภาพรูป)	Flow chart ของ Code ตัวอย่าง
<pre> PS D:\VSCode> cd "d:\VSCode\" ; if (\$? ?) { gcc pointer.c -o pointer } ; if (\$?) { .\pointer } *a = 10 *a = 20 *a = 30 var = 30 PS D:\VSCode> </pre>	<pre> graph TD Start([Start]) --> Var[*a, var] Var --> A1[a = &var] A1 --> A2[*a] A2 --> Var2[var = 30] Var2 --> A3[*a] A3 --> Var3[var] Var3 --> Stop([Stop]) A1 --> A2 A2 --> A3 A3 --> Stop </pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ซึ่ง Array ไม่ใช้ก็แมตร แต่จะ 4 Column และบอกรือว่าการใช้งาน พร้อม Code ด้านล่าง ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>ต้องการที่ต้องการสร้าง Array เพื่อมาเก็บต่อไป จะเพิ่งไปให้ไปต้องมากรอกขนาดของ array ให้ตั้งแต่ 0 ต้อง Array จะเพิ่งจะเรียบๆ ไม่ต้องใส่ตัวอักษร</p>	<pre> 1 #include<stdio.h> 2 3 int main() 4 { 5 int g[3][4] = 6 { 7 { 10, 20, 30, 40 } , 8 { 11, 21, 31, 41 } , 9 { 12, 22, 32, 42 } 10 }; 11 12 int (*j)[4] = g; 13 14 j[0][3] = 99; 15 for(int a = 0 ; a < 3 ; a++) 16 { 17 for(int b = 0 ; b < 4 ; b++) 18 { 19 printf("%d ", g[a][b]); 20 }//end for 21 printf("\n"); 22 }//end for 23 return 0 ; 24 }</pre>

ผลลัพธ์ของ Code (Captureหน้าจอประจุ)	Flow chart ของ Code ตัวอย่าง
<pre> PS D:\VSCode> cd "d:\VSCode\" ; if (\$?) { gcc PA.c -o PA } ; if (\$?) { .\PA A } 10 20 30 99 11 21 31 41 12 22 32 42 PS D:\VSCode></pre>	<pre> graph TD Start((Start)) --> InitG[g[3][4] = {10,20,30,40}, {11,21,31,41}, {12,22,32,42}] InitG --> InitJ[int(*j)[4] = g] InitJ --> SetJ[j[0][3] = 99] SetJ --> A((A)) A --> Cond1{a < 3 ?} Cond1 -- No --> Stop((Stop)) Cond1 -- Yes --> A0[a = 0] A0 --> AA[a++] AA --> Cond2{b < 4 ?} Cond2 -- No --> Stop Cond2 -- Yes --> B0[b = 0] B0 --> G[g[a][b]] G --> BB[b++] BB --> Cond2 </pre>

ข้อที่ 3 จاإรีบานหื่อ Pointer Function ยกเว้นตัวฟังก์ชัน พิมพ์ Code ที่วิเคราะห์มาให้ใช้งาน

คำอธิบาย

คือการหื่อPointer ไปยัง address ของ Function
ที่ส่งเข้ามา และสามารถเรียกใช้งาน function นั้น
ตามที่นั่นได้

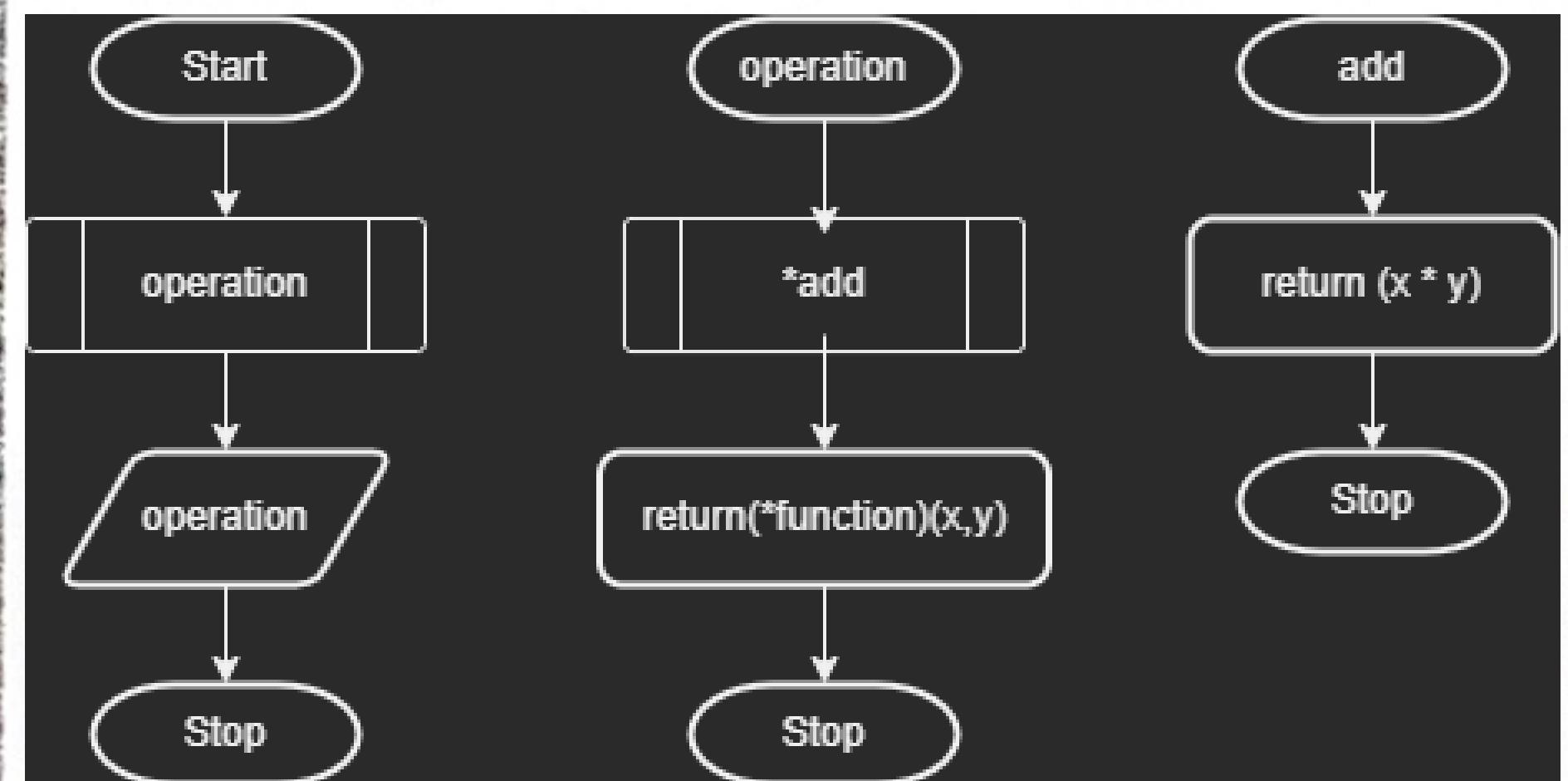
Code ตัวอย่าง

```
1 #include<stdio.h>
2
3 int add( int x, int y ){
4     return x + y ;
5 }
6
7 int operation( int x, int y, int (*function) ( int, int ) ){
8     return (*function)( x, y );
9 }
10
11 int main()
12 {
13     printf("==> %d\n" , operation( 5, 5, add ) );
14     return 0 ;
15 }
```

ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)

```
PS D:\VSCode> cd "d:\VSCode\" ; if ($?
?) { gcc PF.c -o PF } ; if ($?) { ./P
F }
==> 10
PS D:\VSCode>
```

Flow chart ของ Code ศึกษา



ข้อที่ 1 จاإธิบายเพื่อ Dynamic Array ยกตัวอย่างการใช้งาน พัฒนา Code ตัวอย่างการใช้งาน

คำอธิบาย

คือ การสร้าง Array ที่ไม่ระบุขนาดตามตัว โดยจะเพิ่มเติม
ขนาดใหม่ ทั้งนี้ต้องรู้ว่ามีกี่แถว กี่列 เพื่อมาจัดตำแหน่ง
ตามที่ต้องการ แต่เพิ่มได้บันทึกช่องที่ได้รับมา

Code ตัวอย่าง

```

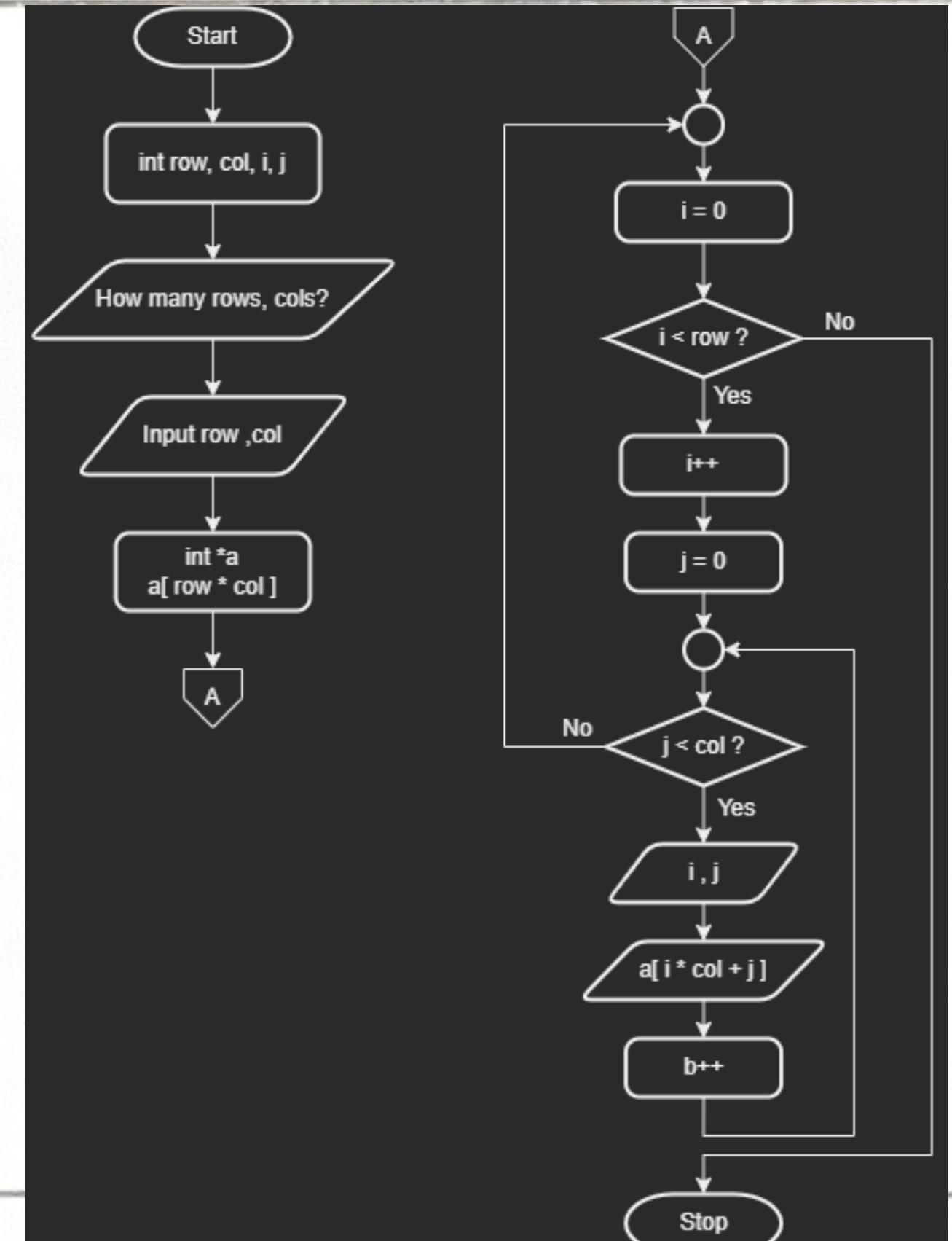
1 #include <stdio.h>
2
3 int main()
4 {
5     int row, col, i, j;
6     printf("How many rows, cols? ");
7     scanf("%d %d", &row, &col); //Input max array[row][column]
8
9     int *a;
10    a[ row * col ];
11
12    for(int i = 0 ; i < row ; i++)
13    {
14        for(int j = 0 ; j < col ; j++)
15        {
16            printf("a[%d][%d]: ", i, j);
17            scanf("%d", &a[i * col + j] );
18        } //end for
19    } //end for
20
21    return 0 ;
22 }
```

ผลลัพธ์ของ Code (Capture หรือแม่รูป)

```

PS D:\VSCode> cd "d:\VSCode\" ; if ($?
?) { gcc DA.c -o DA } ; if ($?) { ./DA
A }
How many rows, cols? 3 3
a[0][0]: 1
a[0][1]: 2
a[0][2]: 3
a[1][0]: 4
a[1][1]: 5
a[1][2]: 6
a[2][0]: 7
a[2][1]: 8
a[2][2]: 9
PS D:\VSCode>
```

Flow chart ของ Code ตัวอย่าง



ข้อที่ 5 จงอธิบายการส่งค่าผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างมาใช้ในการพิมพ์ Code ด้านล่างมาใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>คือการส่งค่าไปปั้งฟังก์ชันที่ถูกเรียกใช้โดยส่งไปในตัวแหน่งที่อยู่ (Address) ของตัวแปรไปสู่ทางทากำนั้น ฟังก์ชันมีการเปลี่ยนแปลงค่าของตัวแปรที่ส่งไป เมื่อผลลัพธ์ในตัวเอง ตัวแปรที่รับมานั้นก็จะไปเปลี่ยนตัวเอง เรียกว่าเปลี่ยนไปด้วย โจทย์ใช้คำว่า Pointer บ่อ Array</p>	<pre> 1 #include <stdio.h> 2 3 void ar(int **a, int *row, int *col) 4 { 5 printf("How many row, col ?\n"); 6 scanf("%d %d", row, col); 7 *a = new int[*row * *col]; 8 for(int i = 0 ; i < *row ; i++) 9 { 10 for(int j = 0 ; j < *col ; j++) 11 { 12 printf("a[%d][%d] = ", i, j); 13 scanf("%d", &(*a)[i * *col + j]); 14 }//end for 15 }//end for 16 } //end function ar 17 18 void ar(int **a, int *row, int *col); 19 20 int main() 21 { 22 int r, c, *arr; 23 ar(&arr, &r, &c); 24 return 0 ; 25 } //end function </pre>

ผลลัพธ์ของ Code (Capture พร้อมภาพรูป)	Flow chart ของ Code ตัวอย่าง
<pre> PS D:\VSCode> cd "d:\VSCode\" ; if (\$?) { g++ pbr.cpp -o pbr } ; if (\$?) { .\pbr } How many row, col ? 3 2 a[0][0] = 1 a[0][1] = 2 a[1][0] = 3 a[1][1] = 4 a[2][0] = 5 a[2][1] = 6 PS D:\VSCode> </pre>	<pre> graph TD Start([Start]) --> Init[int r, c, *arr] Init --> Ar[ar] Ar --> InputRow[Input row, col] InputRow --> I0[i = 0] I0 --> CondI{i < *row} CondI -- No --> Stop([Stop]) CondI -- Yes --> J0[j = 0] J0 --> CondJ{j < col ?} CondJ -- No --> Stop CondJ -- Yes --> Print[a[i][j]] Print --> Input[Input &a[i * col + j]] Input --> Jplus1[j++] Jplus1 --> CondI </pre>

ข้อที่ ๖ จ่ายอธิบายการส่งค่าแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างมาใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>คือการ ส่งค่ากลับค่า ที่เก็บจากตัวแปรที่ส่งไปมายัง ไปร่องฟังก์ชัน โดยผ่านการ return ให้กับ</p>	<pre> 1 #include <stdio.h> 2 3 int *ar(int *row, int *col) 4 { 5 int *a; 6 printf("How many row, col ?\n"); 7 scanf("%d %d", row, col); //Input array 8 a = new int[*row * *col]; 9 for(int i = 0 ; i < *row ; i++) 10 { 11 for(int j = 0 ; j < *col ; j++) 12 { 13 printf("a[%d][%d] = ", i, j); 14 scanf("%d", &a[i * *col + j]); 15 } //end for 16 } //end for 17 return a; 18 } //end function ar 19 20 int *ar(int *row, int *col); 21 22 int main() 23 { 24 int r, c, *arr; 25 arr = ar(&r, &c); 26 delete [] arr; 27 return 0; 28 } //end function </pre>

ผลลัพธ์ของ Code (Capture ภาพรีบูต)	Flow chart ของ Code ตัวอย่าง
<pre> PS D:\VSCode> cd "d:\VSCode\" ; if (\$? ?) { g++ pbv.cpp -o pbv } ; if (\$?) { .\pbv } How many row, col ? 3 2 a[0][0] = 1 a[0][1] = 2 a[1][0] = 3 a[1][1] = 4 a[2][0] = 5 a[2][1] = 6 PS D:\VSCode> </pre>	<pre> graph TD Start([Start]) --> InitInt[int r, c, *arr] InitInt --> CreateAr[ar] CreateAr --> DeleteArr[delete[] arr] DeleteArr --> Ar([ar]) Ar --> InputRowCol[/Input row, col/] InputRowCol --> CreateArr[arr[*row * *col]] CreateArr --> InitI[i = 0] InitI --> A((A)) A --> CondI{i < *row} CondI -- No --> Stop([Stop]) CondI -- Yes --> CondJ{j < *col} CondJ -- No --> ReturnArr([return arr]) CondJ -- Yes --> J0[j = 0] J0 --> AJ[a[i][j]] AJ --> InputArr[/Input &a[i * *col + j]/] InputArr --> Jplus1[j++] Jplus1 --> CondJ </pre>