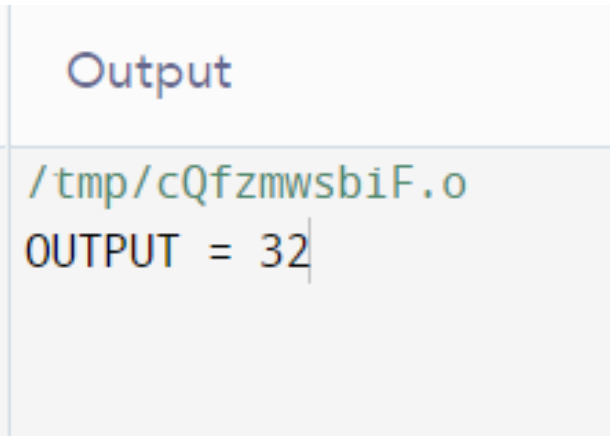
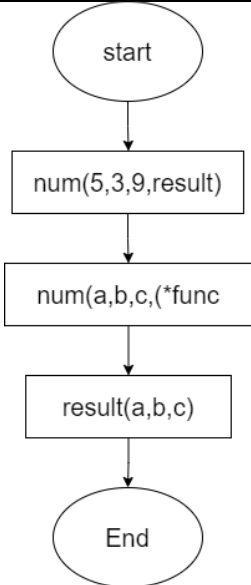


ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>Pointer คือตัวชี้ค่าเก็บค่าที่อยู่ของหน่วยความจำ Dynamic data structure ตัวแปรชนิด Pointer จะเก็บค่าที่อยู่ของหน่วยความจำ รูปแบบการเขียน int *variable</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 (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>Output</p> <pre>/tmp/vpdzqgjYS5.o Enter your value : 5 ---OUTPUT--- *p = 5 v = 5</pre>	<pre>graph TD START([START]) --> PV[p, v] PV --> PAV[p = &v] PAV --> V[/v/] V --> P{{p}} P --> V{{v}} V --> END([END])</pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>สร้าง array 2 มิติแบบปกติขึ้นมาและกำหนดค่าลงไปและใช้ for loop ร้องกันในการ printf แถวละ 4 คอลัมน์ใน array 2 มิติ และประกาศใช้ array ไม่จำกัดแถว รับรูปแบบ (*n)[5] = c; โดยที่ c ว่าเป็นค่า (*n) ี่รับหรือจะขอเปลี่ยนค่าใน array 2 มิติใช้ for loop printf ค่าออกมาเหมือนเดิมโดยการใช้ c[a][b] แล้วจึงเพราะกำหนดค่าให้ตัวแปรก่อนจึงจะป้อนค่า</p>	<pre> main.c 1 #include <stdio.h> 2 int main(){ 3 int a,b; 4 int c[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 ", c[a][b]); 13 printf("\n"); 14 } 15 16 printf("-----\n"); 17 18 int (*n)[5] = c; 19 n[3][3] = 89; 20 n[3][0] = 70; 21 22 for(a = 0; a < 4; a++){ 23 for(b = 0; b < 5; b++){ 24 printf("%d ", c[a][b]); 25 printf("\n"); 26 } 27 } 28 return 0; </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>Output</p> <pre> /tmp/cJzA0zKatK.o 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 </pre>	<pre> graph TD Start([start]) --> Init["a,b = 0; c[4][5] = { {10,20,30,40,50}, {11,21,31,41,51}, {12,22,32,42,52}, {13,23,33,43,53} }"] Init --> LoopA{a < 4} LoopA -- N --> End([End]) LoopA -- Y --> LoopB{b < 5} LoopB -- N --> PrintRow["printf(\"%d \", c[a][b]);"] PrintRow --> PrintSep["printf(\"-----\");"] PrintSep --> DeclN["int (*n)[5] = c;"] DeclN --> AssignN["n[3][3] = 89; n[3][0] = 70;"] AssignN --> LoopA2{a < 4} LoopA2 -- N --> End LoopA2 -- Y --> LoopB2{b < 5} LoopB2 -- N --> PrintRow2["printf(\"%d \", c[a][b]);"] PrintRow2 --> PrintSep2["printf(\"-----\");"] PrintSep2 --> LoopB2 LoopB2 -- Y --> LoopA2 LoopA2 -- Y --> LoopB2 </pre>

ข้อที่ 3 จงอธิบายเรื่อง Pointer Functionยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
เริ่มแรกเราต้องได้ค่าจาก main ก่อนเพื่อ ส่งไปให้ฟังก์ชันของเรา Pointer ไม่ได้ ตัวแปรได้ค่าจากตัวแปรที่เราส่งค่าให้ บางครั้งเราอาจ argument ตัวไม่ต้องการชื่อ ตัวแปรตัวไปก็ได้และเราสามารถนำผลไป กับตัวอื่นที่รับกันได้	<pre>#include <stdio.h> int result (int a,int b,int c){ 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; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre>graph TD Start([start]) --> Num1[num(5,3,9,result)] Num1 --> Num2[num(a,b,c,(*func] Num2 --> Result[result(a,b,c)] Result --> End([End])</pre>

ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>เป็นตัวแปร array ที่ใช้รับข้อมูลมา คือแปรที่ไม่กำหนดค่าแน่นอน แต่ต้องการ ใช้ array แบบ Dynamic เพื่อประหยัดทรัพยากร ระบบโดยที่ตัวแปร array จะปรับขนาด ให้เหมาะสมกับค่าที่รับมาคือแปรที่ต้องการ ใช้ในขณะที่นั้น จะเน้นได้ว่าสามารถใช้ทรัพยากรระบบ เมื่อต้องการเท่านั้น จึงสิ้นเปลืองอย่างต่ำ</p>	<pre>#include <stdio.h> int main () { int row, col, i, j; printf("How many rows, cols? "); scanf("%d %d", &row, &col); int *c; c = 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; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>Output</p> <pre>/tmp/L0EjFrEY1C.o How many rows, cols? 2 3 a[0][0]: 3 a[0][1]: 4 a[0][2]: 7 a[1][0]: 8 a[1][1]: 9 a[1][2]: 6</pre>	<pre>graph TD Start([start]) --> Init[row, col, i, j] Init --> Input[/row, col/] Input --> Alloc[b = new int [row * col]] Alloc --> Loop1{i < row} Loop1 -- N --> Input Loop1 -- Y --> Loop2{j < col} Loop2 -- N --> Loop1 Loop2 -- Y --> Input2[/b[i*col+j]/] Input2 --> IncJ[j+1] IncJ --> IncI[i+1] IncI --> Loop1 IncI --> End([End])</pre>

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Pass by reference โดยที่เอาค่าของ Argument และ Argument ที่ถูกแก้ไขมาจัดเก็บไว้ใน array ก็เป็น Pointer คือใน array ข้างนอกเป็นเหมือน Pointer 1 ตัวอยู่แล้วก็จะเอาข้อมูลออกจะได้อีกมากทำให้ง่าย 1 ตัว</p>	<pre> main.cpp 1 #include <stdio.h> 2 3 void GetArr2D(int **a,int *row,int *col); 4 5 int main(){ 6 int r, c, *arr ; 7 GetArr2D(&arr, &r, &c); 8 return 0 ; 9 } 10 11 void GetArr2D(int **a,int *row,int *col){ 12 int i, j; 13 printf("How many row, col? "); 14 scanf("%d %d", row, col); 15 *a = new int[*row * *col]; 16 for(i = 0; i < *row; i++){ 17 for(j = 0; j < *col; j++){ 18 printf("a[%d][%d] = ", i, j); 19 scanf("%d", &(*a)[i * *col + j]); 20 } 21 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>Output</p> <pre> /tmp/Ygsr61sK1A.o How many row, col? 2 3 a[0][0] = 4 a[0][1] = 7 a[0][2] = 8 a[1][0] = 9 a[1][1] = 6 a[1][2] = 3 </pre>	<pre> graph TD Start([start]) --> Init[r,c, *arr] Init --> Call1[GetArr2D(&arr, &r, &c)] Call1 --> Call2[GetArr2D(**a,*row,*col)] Call2 --> End([End]) </pre>

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p style="font-size: 1.2em; color: blue;">Pass by Value คือ การเอาค่าออกไปใช้ 2 ครั้ง</p> <p style="font-size: 1.2em; color: blue;">Pass by Value 1 ครั้ง และ by Reference 2 ครั้ง</p>	<pre style="background-color: #f0f0f0; padding: 10px;">main.cpp 1 #include <stdio.h> 2 3 int *GetArr2D(int *row, int *col) ; 4 5- int main() { 6 int r, c, *arr ; 7 arr = GetArr2D(&r, &c) ; 8 delete [] arr; 9 return 0; 10 } 11- int *GetArr2D(int *row, int *col) { 12 int *a ; 13 int i, j; 14 printf("How many row, col? ") ; scanf("%d %d", row, col); 15 a = new int[*row * *col]; 16 for(i = 0; i < *row ; i++) 17 for(j = 0; j < *col; j++) { 18 printf("a[%d][%d] = ", i, j); 19 scanf("%d", &a[i * *col + j]) ; 20 } 21 return a; 22 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<div style="border: 1px solid #ccc; padding: 10px; background-color: #f9f9f9;"> <p style="color: #4a7ebb; font-weight: bold;">Output</p> <pre style="margin: 0;">/tmp/GUCKdB9zkL.o How many row, col? 2 3 a[0][0] = 6 a[0][1] = 3 a[0][2] = 2 a[1][0] = 1 a[1][1] = 4 a[1][2] = 7</pre> </div>	<pre> graph TD Start([start]) --> Init[r,c, *arr] Init --> Call[arr = GetArr2D(&arr, &r, &c)] Call --> Call2[*GetArr2D(*row,*col)] Call2 --> Delete[delete [] arr] Delete --> End([End]) </pre>