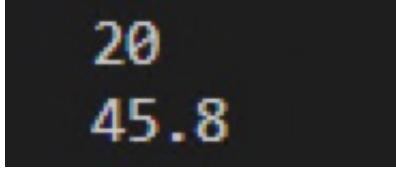
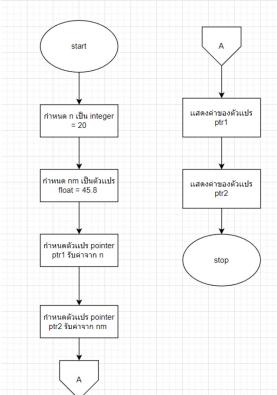


ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวัดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

คำอธิบาย	Code ตัวอย่าง
<p>Pointer คือ ตัวที่คุณเขียนให้เป็นตัวที่อยู่ทางข้างหลัง หรือในทางภาษาพื้นเมือง โลกใช้ภาษาทันสมัยมากกว่า</p> <p>โดยที่ตัวของตัวที่อยู่ในในตัว pointer ไม่แน่</p> <p>Address หรือที่อยู่ (&)</p> <p>Dereference Operator ตัวที่เพิ่มลงมาหนึ่งชั้น (*)</p> <p>Declaration operator ตัวที่เพิ่มลงมาสองชั้น (**)</p>	<pre> 1 #include<stdio.h> 2 3 int main() 4 { 5 int n = 20 ; 6 float nm = 45.8 ; 7 8 int *ptr1 = &n ; 9 float *ptr2 ; 10 ptr2 = &nm ; 11 printf("%d\n", *ptr1) ; 12 printf("%.1f\n", *ptr2) ; 13 return 0 ; 14 } 15 16 17 </pre>
ผลลัพธ์ของ Code (Captureพร้อมแบบรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD start((start)) --> initN[int("กำหนด n เป็น integer = 20")] initN --> initNM[int("กำหนด nm เป็น float float = 45.8")] initNM --> initP1[int("กำหนดตัวแปร pointer ptr1 ให้มีค่าเป็น n")] initP1 --> initP2[int("กำหนดตัวแปร pointer ptr2 ให้มีค่าเป็น nm")] initP2 --> A{A} A --> printP1[int("แสดงผลตัวแปร ptr1 ค่าเท่ากับ n")] printP1 --> stop((stop)) initP2 --> printP2[int("แสดงผลตัวแปร ptr2 ค่าเท่ากับ nm")] printP2 --> A </pre>

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

คำอธิบาย	Code ตัวอย่าง
<p>สร้าง Pointer ชี้ไปยังตัว Array หรือ Array นั้นก็ได้ ตัวนุ่ง เป็นตัวอักษร ไม่สามารถตัวอักษร</p>	<pre> 1 #include<stdio.h> 2 3 int main() 4 { 5 int num[5] = {10, 20, 30, 40, 50}; 6 int *Pointer = num ; 7 8 9 printf("1state -> %d\n" , *Pointer); 10 Pointer++; 11 printf("2state -> %d\n" , *Pointer); 12 Pointer+=3; 13 printf("3state -> %d\n" , *Pointer); 14 Pointer--; 15 printf("4state -> %d\n" , *Pointer); 16 return 0; 17 } 18 19 </pre>
ผลลัพธ์ของ Code (Captureพร้อมภาพรูป)	Flow chart ของ Code ตัวอย่าง
<p>1state -> 10 2state -> 20 5state -> 50 4state -> 40</p>	<pre> graph TD start((start)) --> init["ก่อตั้งเป็นมассив array ขนาด 5 ค่า int 10 20 30 40 50"] init --> pointerInit["ก่อตั้งตัวแปร pointer ให้เป็น null หรือ num"] pointerInit --> print1["state pointer -> %d\n"], A1[/A/] print1 --> pointerInc1["Pointer++"] pointerInc1 --> print2["state pointer -> %d\n"], A2[/A/] print2 --> pointerDec3["Pointer-=3"] pointerDec3 --> print3["state pointer -> %d\n"], A3[/A/] print3 --> pointerDec1["Pointer--"] pointerDec1 --> print4["state pointer -> %d\n"], A4[/A/] print4 --> stop((stop)) </pre>

ข้อที่ 3 จงอธิบายเรื่อง Pointer Function กดตัวอย่างการใช้งาน พิจารณาพร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>ในไฟล์นี้ ห้องเรียน โครงสร้าง หนึ่งตัวฟังก์ชัน คือ ตัวฟังก์ชัน pointer ที่มีรูปแบบ parameter คือ <code>void (*functionname)(parameter)</code></p>	<pre> 1 #include <stdio.h> 2 3 void setup(int a) 4 { 5 printf("pointer value %d\n", a); 6 } 7 8 int main() 9 { 10 void (*setup_ptr)(int) = setup; /* removed 11 setup_ptr(10); /* removed 12 return 0 ; 13 } 14 }</pre>
ผลลัพธ์ของ Code (Captureพิจารณาแบบรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD start((start)) --> pointer[call pointer Function to setup_ptr] pointer --> state[State pointer -> %d] state --> stop((stop)) pointer --> call[call setup_ptr] call --> void[void setup receives int] void --> value[pointer value 10] value --> stop </pre>

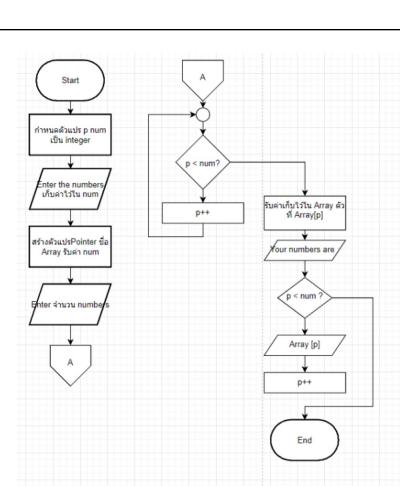
ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Dynamic Array คือ Array ที่สามารถเพิ่มน้ำหนักได้ 9 ในการเขียนภาษา C จะสามารถเพิ่มน้ำหนักได้โดยการเพิ่มพื้นที่ความจำเพิ่มเติม Array Dynamic</p>	<pre> 1 #include<stdio.h> 2 3 int main() 4 { 5 int row,col; 6 printf("Input your row and column: "); 7 scanf("%d %d", &row, &col); 8 9 int *a; 10 a = new int (row*col); 11 for(int i = 0 ; i < row ; i++){ 12 for(int j = 0 ; j < col ; j++){ 13 printf(" a[%d][%d] = " , i,j); 14 scanf("%d" , &a[i * col + j]); 15 } 16 } 17 18 for(int i = 0; i < row * col ; i++){ 19 printf("%d\t" , a[i]); 20 if((i+1)%col == 0){ 21 printf("\n"); 22 } 23 } 24 25 26 27 28 29 30 }</pre>
ผลลัพธ์ของ Code (Captureพิมพ์แบบรูป)	Flow chart ของ Code ตัวอย่าง

```

Input your row and column: 2
2
a[0][0] = 1
a[0][1] = 2
a[1][0] = 3
a[1][1] = 4
1      2
3      4

```



ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by Ref คือ ไม่ต้องcopy值เข้าไปในพารามิเตอร์ โดยส่ง ให้พารามิเตอร์ ของ (Address) ของตัวแปรไปท่องทางภายในฟังก์ชัน แล้ว ฟังก์ชันจะดูแลห้อง Argument ของตัวแปรที่เก็บไปท่องทางไปท่อง</p>	<pre> 1 #include<stdio.h> 2 void swap(int &x , int &y){ 3 int temp; 4 5 temp = x; /* เก็บ値 pointer x ไว้ที่ temp 6 x = y; /* หันค่า pointer y เก็บไว้ที่ pointer x 7 y = temp; /* หันค่า temp เก็บไว้ที่ pointer y เป็นการสลับค่ากัน 8 } 9 10 int main(){ 11 int a = 100; 12 int b = 200; 13 14 printf("Before swap, value of a : %d\n", a); 15 printf("Before swap, value of b : %d\n", b); 16 17 swap(a , b); 18 19 printf("After swap , value of a : %d\n", a); 20 printf("After swap , value of b : %d\n", b); 21 22 return 0 ; 23 } 24 25 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมภาพรูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD Start([Start]) --> DeclTemp[กำหนดค่า temp เป็น integer] DeclTemp --> SetX[กำหนดค่า pointer x =] SetX --> SetY[กำหนดค่า pointer y =] SetY --> Swap[swap(&a,&b)] Swap --> EndTemp[กำหนดค่า temp เป็น integer] EndTemp --> End([End]) </pre>
<pre> Before swap, value of a : 100 Before swap, value of b : 200 After swap , value of a : 200 After swap , value of b : 100 </pre>	

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by value คือ การส่งค่าใน argument ของฟังก์ชัน ไม่ส่งค่ากลับคืน</p> <p>ค่าใน argument จะถูกเปลี่ยนแปลง</p>	<pre> 1 #include<stdio.h> 2 3 void swap(int i_sw , int j_sw) 4 { 5 int temp; 6 7 temp = i_sw; 8 i_sw = j_sw; 9 j_sw = temp; 10 11 return; 12 } 13 int main() 14 { 15 int i = 100; 16 int j = 200; 17 18 printf("Before swap: value of i : %d\n", i); 19 printf("Before swap: value of j : %d\n", j); 20 21 swap(i , j); 22 23 printf("After swap: value of i : %d\n", i); 24 printf("After swap: value of j : %d\n", j); 25 26 return 0; </pre>
ผลลัพธ์ของ Code (Captureพร้อมภาพรูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD Start((start main(void))) --> DeclI[i = 100 j = 200 (2x integer)] DeclI --> BeforeI[before swap: value of i : %d in i] BeforeI --> SwapI[temp = i_sw] SwapI --> DeclTemp[int temp] DeclTemp --> SwapJ[i_sw = j_sw] SwapJ --> BeforeJ[before swap: value of j : %d in j] BeforeJ --> SwapTemp[j_sw = temp] SwapTemp --> AfterSwapI[After swap: value of i : %d in i] AfterSwapI --> AfterSwapJ[After swap: value of j : %d in j] AfterSwapJ --> StopMain((stop main(void))) </pre>

Before swap: value of i : 100
 Before swap: value of j : 200
 After swap: value of i : 100
 After swap: value of j : 200