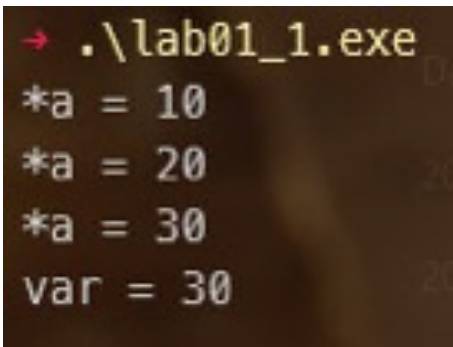
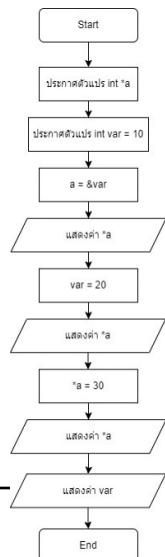

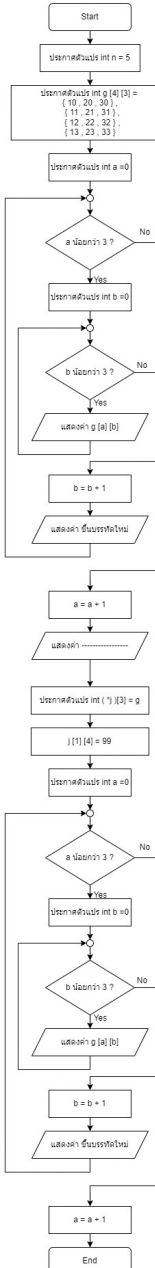


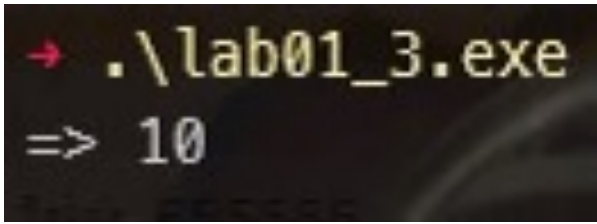
ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

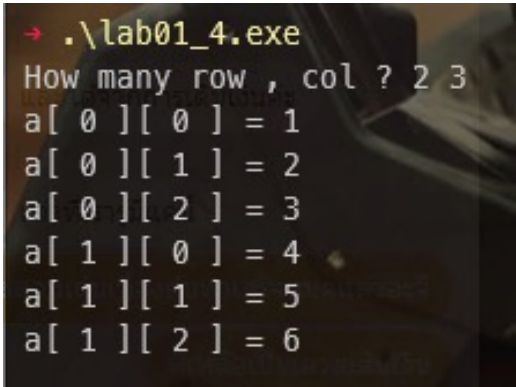
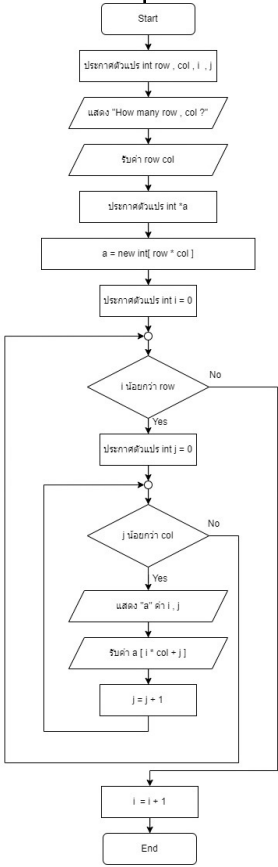
ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>Pointer คือ ตัวชี้ ซึ่งถือเป็นลักษณะเด่นของ Operator ในภาษา C ใช้ในการจัดการข้อมูล Dynamic data structure</p> <p>ตัวแปรชนิด Pointer จะเก็บค่า "ที่อยู่" ของหน่วยความจำ (แตกต่างกับตัวแปรชนิดอื่น ที่เก็บค่าจริง)</p> <p>รูปแบบการประกาศตัวแปร คือ *</p> <p>ชนิดข้อมูล *ชื่อตัวแปร</p> <p>เช่น <code>int *variable;</code></p>	<pre>#include <stdio.h> int main () { int *a; int var = 10; a = &var; print ("%a = %d\n", *a); var = 20; print ("%a = %d\n", *a); *a = 30; print ("%a = %d\n", *a); print ("%var = %d\n", var); return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre>graph TD Start([Start]) --> A[ประกาศตัวแปร int *a] A --> B[ประกาศตัวแปร int var = 10] B --> C[a = &var] C --> D[/แสดงค่า *a/] D --> E[var = 20] E --> F[/แสดงค่า *a/] F --> G[*a = 30] G --> H[/แสดงค่า *a/] H --> I[/แสดงค่า var/] I --> End([End])</pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>ทำการสร้างตัวแปร int Array ซึ่งก 1 ตัว เป็น 2 มิติ โดยให้มี แถวไม่จำกัด แถวละ 4 Column จากนั้นทำการชี้ตัว Array 1 ตัว ในนั้น โดยทำการเปลี่ยนเป็นเลข 99</p>	<pre>#include <stdio.h> int main() { int n = 5; int g[4][3] = { { 10, 20, 30 }, { 11, 21, 31 }, { 12, 22, 32 }, { 13, 23, 33 } }; for (int a = 0; a < 3; a++) { for (int b = 0; b < 4; b++) { printf("%d ", g[a][b]); } //end for printf("\n"); } //end for printf("\n ----- \n"); int (*j)[3] = g; j[1][4] = 99; for (int a = 0; a < 3; a++) { for (int b = 0; b < 5; b++) { printf("%d ", g[a][b]); } //end for printf("\n"); } //end for return 0; } //end function</pre>
<p>ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)</p> 	<p>Flow chart ของ Code ตัวอย่าง</p> 

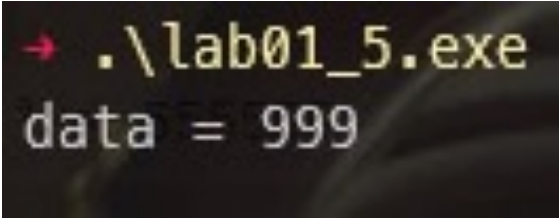
ข้อที่ 3 จงอธิบายเรื่อง Pointer Function ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>เป็นการสร้างฟังก์ชัน Pointer Function ในการเก็บค่าจากฟังก์ชัน main ใส่ใน Array Pointer ไปไว้ในกระบวนการในฟังก์ชันอื่นๆ</p>	<pre>#include <stdio.h> int add (int x , int y){ return x + y ; } int operation (int x, int y, int (*function)(int,int)){ return (*function)(x, y); } int main() { printf("=> %d \n", operation(5,5,add)); return 0 ; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD subgraph main StartMain([Start]) --> Display[/แสดง "=>" ค่าของ operation(5,5,add)/] Display --> EndMain([End]) end subgraph operation_function [operation function] StartOp([Start]) --> ReceiveOp[รับค่า x และ y และ int(*function) , (x , y)] ReceiveOp --> SendOp[ส่งค่า x + y กลับ] SendOp --> EndOp([End]) end subgraph add_function [add function] StartAdd([Start]) --> ReceiveAdd[รับค่า x และ y] ReceiveAdd --> SendAdd[ส่งค่า x + y กลับ] SendAdd --> EndAdd([End]) end </pre>

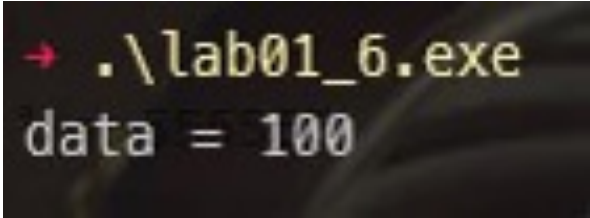
ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Dynamic Array สามารถเอาไปทำเป็น Array ที่มีลิสต์ที่ตัวก็ได้โดยการเอาไปจองหน่วยความจำ เช่น <code>int *b b=new int[n];</code> เอา b ไปลอง Address</p> <p>ภรรมา จง 2 มิติ ได้ ข้อสังเกต Pointer ขนาดหน่วยข้อมูลไม่ใช่</p>	<pre>#include <stdio.h> int main () { int row , col , i , j ; printf ("How many rows , cols ? "); scanf ("%d %d" , &row , &col); int *a ; a = new int [row * col]; for (int i = 0 ; i < row ; i++) { for (int j = 0 ; j < col ; j++) { printf ("a [%d] [%d] : " , i , j); scanf ("%d" , &a [i * col + j]); } //end for } //end for return 0 ; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by reference คือการส่งตัวแปร เป็น argument ของฟังก์ชัน ดังนั้น ตัวแปรที่มีค่าเดิมในฟังก์ชัน จะส่งผลให้ตัวแปรนอกฟังก์ชันมีค่าเปลี่ยนแปลงด้วย</p>	<pre>#include <stdio.h> int Data(int *ptr) { *ptr = 999 ; } // end Function int main () { int data = 100; Data (&data) ; print ("data = %d", data) return 0 ; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปรรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD subgraph main Start1([Start]) --> Init1[ประกาศตัวแปร int data = 100] Init1 --> Call1[Data(&data)] Call1 --> Print1[/แสดงค่า "data = "/] Print1 --> End1([End]) end subgraph Data_function [Data function] Start2([Start]) --> Init2[ประกาศตัวแปร int *ptr] Init2 --> Assign2[*ptr = 999] Assign2 --> End2([End]) end </pre>

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Pass by value คือการส่งค่า เป็น argument ของฟังก์ชัน ดังนั้นค่าที่ส่งให้ฟังก์ชันจะไม่ส่งผลกระทบต่อตัวแปรของฟังก์ชัน</p>	<pre>#include <stdio.h> int Data(int data) { data = 999; } // end Function int main () { int data = 100; Data(data); printf("data = %d", data); return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD subgraph main Start1([Start]) --> Print1[ประกาศตัวแปร int data = 100] Print1 --> Call1[Data(data)] Call1 --> Display1[/แสดงค่า "data = "/] Display1 --> End1([End]) end subgraph Data_function [Data function] Start2([Start]) --> Print2[ประกาศตัวแปร int data] Print2 --> Assign2[data = 999] Assign2 --> End2([End]) end </pre>