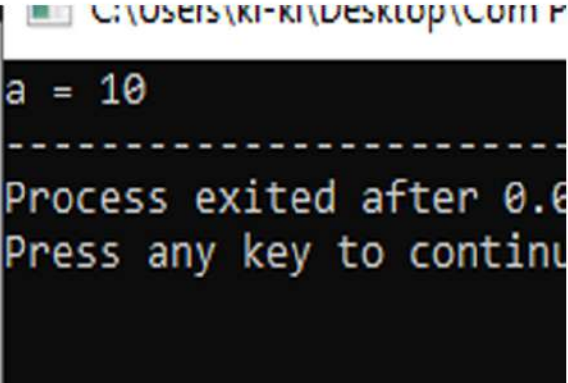
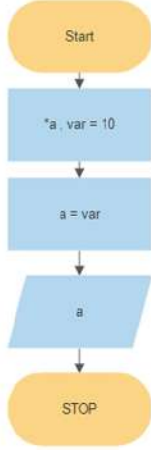
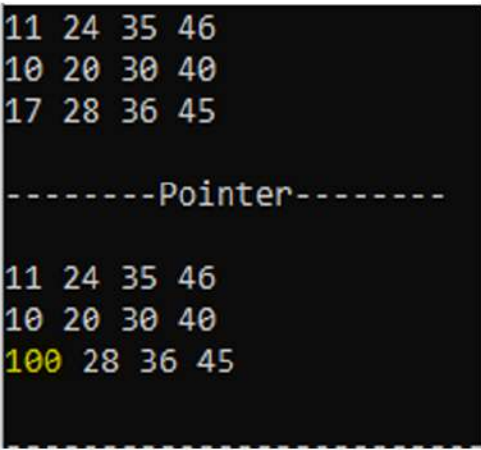
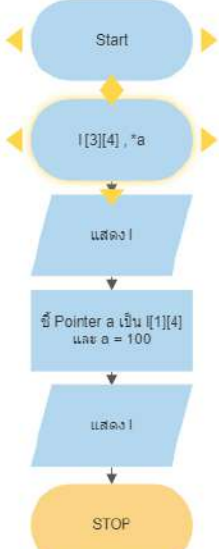
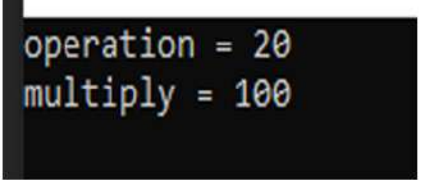
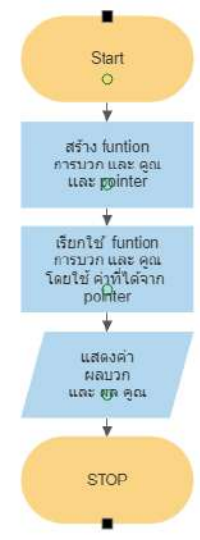


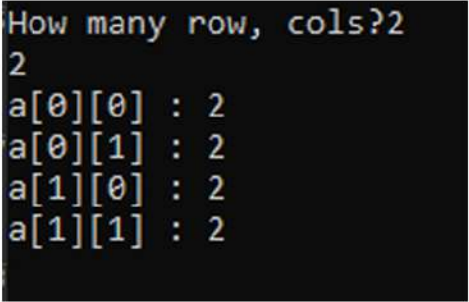
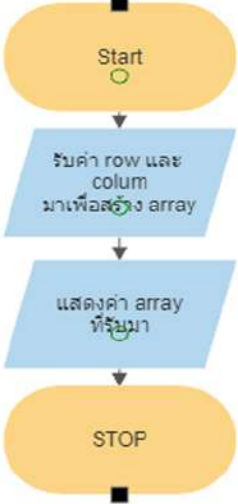
ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

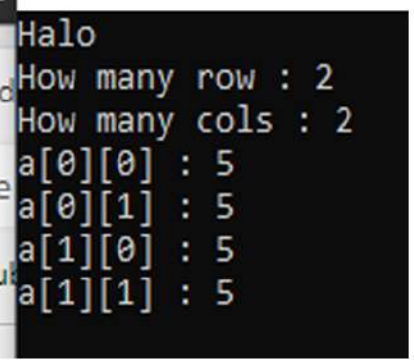

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

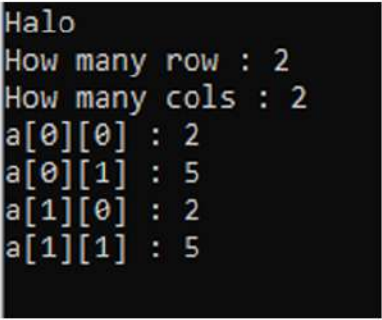

คำอธิบาย	Code ตัวอย่าง
<p>คือการสร้างที่อยู่ให้ข้อมูล มีประโยชน์ในการจัดการข้อมูล ตัวแปรชนิดนี้จะเก็บค่าที่อยู่ของหน่วยความจำต่างกับตัวแปรชนิดอื่นที่เก็บค่าจริง</p>	<pre> 1 #include <stdio.h> 2 3 int main(){ 4 int *a; 5 int var = 10; 6 a = &var; 7 8 printf("a = %d",*a); 9 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD Start([Start]) --> Init[a, var = 10] Init --> Assign[a = var] Assign --> Print[/a/] Print --> Stop([STOP]) </pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>ใน Code ได้ประกาศ array 3 แถว 4 column และดำเนินการประกาศออกก่อน จากนั้นได้ทำการให้ Pointer ชี้ตัว array ที่ แถว 3 ตัวที่ 1 ให้เป็น 100</p>	<pre>#include <stdio.h> int main(){ int l [3][4]= { { 11 , 24 , 35 , 46 }, { 10 , 20 , 30 , 40 }, { 17 , 28 , 36 , 45 } }; for(int a = 0 ; a < 3 ; a++){ for (int b = 0 ; b < 4 ; b++) printf("%d ", l[a][b]); printf("\n"); } printf("\n-----Pointer-----"); printf("\n\n"); int (*a)[4]= l; a[1][4] = 100; for(int a = 0 ; a < 3 ; a++){ for (int b = 0 ; b < 4 ; b++) printf("%d ", l[a][b]); printf("\n"); } return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
 <pre>11 24 35 46 10 20 30 40 17 28 36 45 -----Pointer----- 11 24 35 46 10 20 30 40 100 28 36 45</pre>	 <pre>graph TD Start([Start]) --> Init([l[3][4], *a]) Init --> Show1[/แสดง l/] Show1 --> Point[ชี้ Pointer a เป็น l[1][4] และ a = 100] Point --> Show2[/แสดง l/] Show2 --> Stop([STOP])</pre>

ข้อที่ 3 จงอธิบายเรื่อง Pointer Functionยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Pointer Funtion คือการที่เราสร้าง ชุดคำสั่งไว้เพื่อให้เราเรียกใช้งาน เพื่อลดความยาวของ code และช่วย ให้เราเข้าใจ code ได้ง่ายขึ้นโดยใน code ได้ สร้างขึ้นมา 3 funtion โดยมี funtion บวก , funtion คูณ และ funtion ของตัว Pointer</p>	<pre> 1 #include <stdio.h> 2 3 4 int add(int x,int y){ 5 return x + y; 6 } 7 8 int multiply (int x,int y){ 9 return x * y; 10 } 11 12 int operaion(int x, int y, int (*a) (int,int)){ 13 return (*a)(x,y); 14 } 15 16 int main(){ 17 18 printf("operation = %d\n", operaion(10,10, add)); 19 printf("multiply = %d\n", operaion(10,10, multiply)); 20 21 return 0; 22 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD Start([Start]) --> CreateFunc[สร้าง function การบวก และ คูณ และ pointer] CreateFunc --> UseFunc[เรียกใช้ function การบวก และ คูณ โดยใช้ ตัวที่ใส่จาก pointer] UseFunc --> Display[แสดงค่า ผลบวก และ คูณ] Display --> Stop([STOP]) </pre>

ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Dynamic array คือ array ที่ไม่ได้กำหนดขนาดตายตัว เพราะอาจไม่รู้ว่าจะต้องใช้ขนาดเท่าไรเราจะใช้ Pointer เป็นตัวกำหนด</p> <p>ในตัวอย่าง code จะเป็นการสร้าง ตารางข้อมูล โดยการเก็บข้อมูลมาสร้างเป็น array.....</p>	<pre> 1 #include <stdio.h> 2 3 int main(){ 4 int row, col, i, j; 5 printf("How many row, cols?"); 6 scanf("%d %d", &row, &col); 7 8 int *a; 9 a = new int[row*col]; 10 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", 15 &a[i*col+j]); 16 } 17 return 0; 18 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD Start([Start]) --> Process[รับค่า row และ column มาเพื่อสร้าง array] Process --> Output[/แสดงค่า array ที่รับมา/] Output --> Stop([STOP]) </pre>

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p><u>การส่งตัวแปรแบบ Pass by reference</u></p> <p><u>คือ การส่งค่าไปยังฟังก์ชันที่ถูกเรียกใช้</u></p> <p><u>โดยส่งเป็นค่าตัวแปรที่อยู่ที่ของตัวแปรไม่</u></p> <p><u>ตัวอย่างใน code ได้ทำการรับค่ามาจาก</u></p> <p><u>นั้นได้ทำการส่งค่าไปในฟังก์ชัน</u></p>	<pre> #include <stdio.h> void GetArr2D(int **a, int *row, int *col){ printf("Halo \n"); printf("How many row : "); scanf("%d", row); printf("How many cols : "); scanf("%d", col); *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]); } } void GetArr2D(int **a, int *row,int *col); int main(){ int r,c,*arr; GetArr2D(&arr,&r,&c); return 0; } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD Start([Start]) --> Process[รับค่า row และ column มาเพื่อสร้าง array] Process --> Output[/แสดงค่า array ที่รับมา/] Output --> Stop([STOP]) </pre>

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การส่งตัวแปรแบบ Pass by Value คือ</p> <p>การส่งค่าของฟังก์ชัน ดังนั้นค่าที่ทำในฟังก์ชันจึงไม่ส่งผลต่อตัวแปรนอกฟังก์ชัน</p>	<pre> int *GetArr(int *row, int *col){ int *a; printf("Halo \n"); printf("How many row : "); scanf("%d", row); printf("How many cols : "); scanf("%d", col); 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]); } } return a; } int *GetArr(int *row,int *col); int main(){ int r,c,*arr; arr = GetArr(&r , &c); delete [] arr; return 0; } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD Start([Start]) --> Input[/รับค่า row และ column มาเพื่อสร้าง array/] Input --> Process[/แสดงค่า array ที่รับมา/] Process --> Stop([STOP]) </pre>