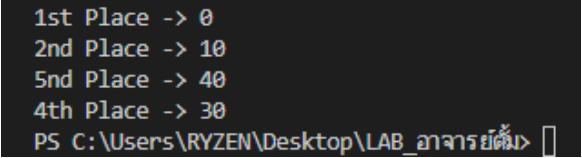
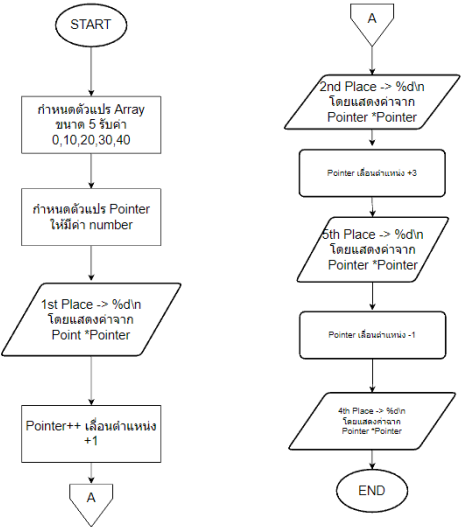
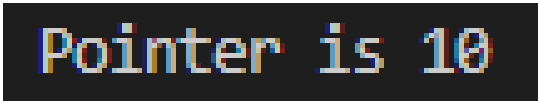
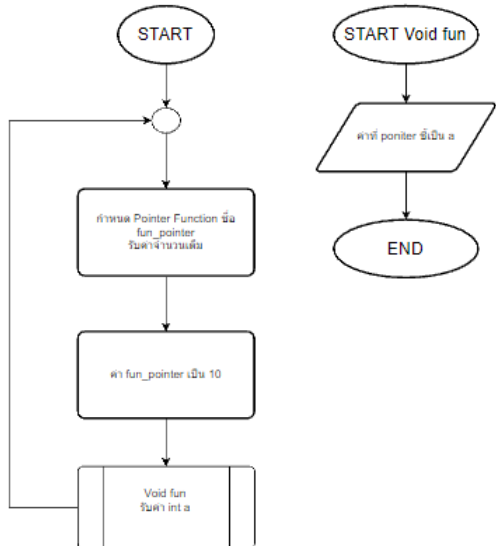
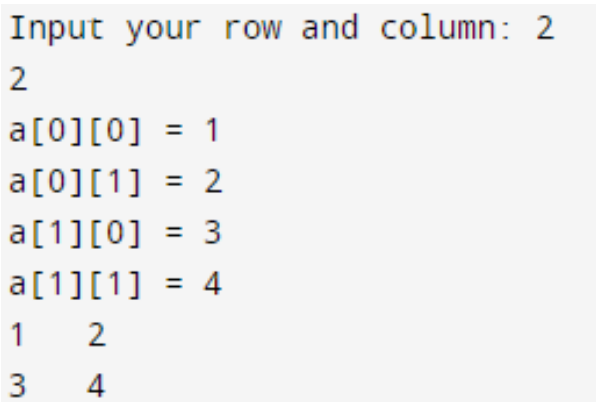
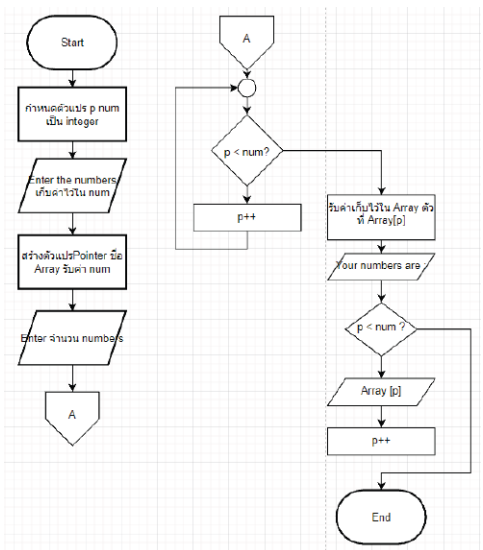
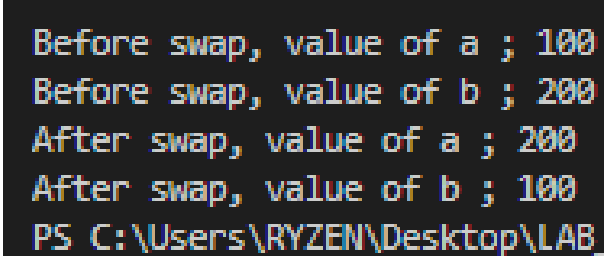
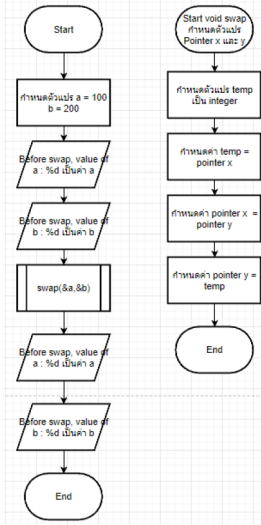


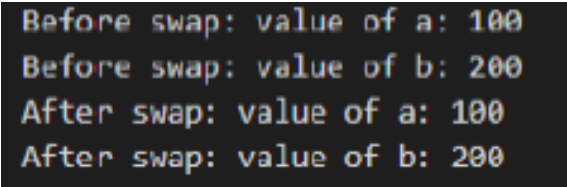
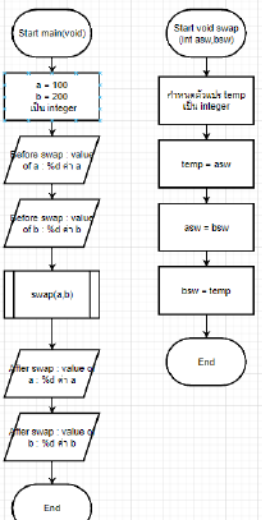
| ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน | |
|---|--|
| คำอธิบาย | Code ตัวอย่าง |
| <p>จากโค้ดเป็นการใช้ Pointer กับ ตัวแปรอาร์เรย์ซึ่ง</p> <p>ด้วยคำสั่ง Pointer ทำให้เราสามารถเลื่อนตำแหน่ง</p> <p>ของค่าที่ถูกจัดเรียงตามลำดับใน Array ได้</p> <p>โดยสามารถเลื่อนตำแหน่งได้โดยการ + หรือ - ค่า</p> <p>Pointer จะแสดงผลพริ้นในตำแหน่งของ Array</p> <p>ที่เราต้องการได้</p> | <pre> C: > Users > RYZEN > Desktop > LAB_อาจารย์คิม > C LAB_1_2.c > main() 1 #include<stdio.h> 2 3 int main(){ 4 int number[5] = {0, 10, 20, 30, 40}; 5 int *Pointer = number; 6 7 printf("1st Place -> %d\n", *Pointer); 8 Pointer++; 9 printf("2nd Place -> %d\n", *Pointer); 10 Pointer += 3; 11 printf("5nd Place -> %d\n", *Pointer); 12 Pointer--; 13 printf("4th Place -> %d\n", *Pointer); 14 return 0; 15 } </pre> |
| ผลลัพธ์ของ Code (Captureพร้อมแปะรูป) | Flow chart ของ Code ตัวอย่าง |
|  |  |

| ข้อที่ 3 จงอธิบายเรื่อง Pointer Functionยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน | |
|--|--|
| คำอธิบาย | Code ตัวอย่าง |
| <p>เป็นฟังก์ชันที่มีโครงสร้างคล้ายๆฟังก์ชันอื่นๆ</p> <p>แต่ประกาศชื่อฟังก์ชันเป็น Pointer โดยต้องมีวงเล็บ</p> | <pre> C: > Users > RYZEN > Desktop > LAB_อาจารย์ดี > C LAB_1_3.C 1 #include <stdio.h> 2 3 void fun(int a) 4 { 5 printf("Pointer is %d\n", a); 6 } 7 8 int main() 9 { 10 void (*fun_pointer)(int) = fun; 11 fun_pointer(10); 12 return 0; 13 } </pre> |
| ผลลัพธ์ของ Code (Captureพร้อมแปะรูป) | Flow chart ของ Code ตัวอย่าง |
|  |  |

| ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน | |
|---|---|
| คำอธิบาย | Code ตัวอย่าง |
| <p>Dynamic Array คืออาร์เรย์ที่สามารถเปลี่ยนขนาดได้ในขณะที่โค้ดกำลังทำงานและสามารถเพิ่มองค์ประกอบได้อย่างต่อเนื่องที่ตำแหน่งล่าสุด</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 12 for(int i = 0; i < row; i++){ 13 for(int j = 0; j < col; j++){ 14 printf("a[%d][%d] = ",i,j); 15 scanf("%d", &a[i * col + j]); 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 return 0 ; 25 }</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 Ref เป็นการส่งค่าไปยังฟังก์ชันที่ถูกเรียกใช้ โดยส่งเป็นค่าตำแหน่งที่อยู่(Address)ของตัวแปร ไม่ซึ่งหากภายในฟังก์ชันมีการเปลี่ยนค่า ก็จะทำให้ค่านั้นเปลี่ยนไปด้วย</p> | <pre> 1 #include <stdio.h> 2 3 ~ int main(){ 4 int a = 100; 5 int b = 200; 6 7 printf("Before swap, value of a ; %d\n", a); 8 printf("Before swap, value of b ; %d\n", b); 9 10 swap(&a, &b); 11 12 printf("After swap, value of a ; %d\n", a); 13 printf("After swap, value of b ; %d\n", b); 14 15 return 0; 16 } 17 ~ void swap(int *x, int *y){ 18 19 int temp; 20 temp = *x; 21 *x = *y; 22 *y = temp; 23 return; 24 }</pre> |
| ผลลัพธ์ของ Code (Captureพร้อมแปะรูป) | Flow chart ของ Code ตัวอย่าง |
|  <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 PS C:\Users\RYZEN\Desktop\LAB</pre> |  |

| ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน | |
|--|--|
| คำอธิบาย | Code ตัวอย่าง |
| <p>Pass by value เป็นการส่งค่าเป็น argument ของ function โดยค่าในฟังก์ชันจะไม่ส่งผลต่อตัวแปรนอกฟังก์ชัน</p> | <pre> 1 void swap(int asw, int bsw) 2 { 3 int temp; 4 5 temp = asw; 6 asw = bsw; 7 bsw = temp; 8 9 return; 10 } 11 12 int main(void) 13 { 14 int a = 100; 15 int b = 200; 16 17 printf("Before swap: value of a: %d\n", a); 18 printf("Before swap: value of b: %d\n", b); 19 20 swap(a, b); 21 22 printf("After swap: value of a: %d\n", a); 23 printf("After swap: value of b: %d\n", b); 24 25 return 0; 26 } </pre> |
| ผลลัพธ์ของ Code (Captureพร้อมแปะรูป) | Flow chart ของ Code ตัวอย่าง |
|  <pre> Before swap: value of a: 100 Before swap: value of b: 200 After swap: value of a: 100 After swap: value of b: 200 </pre> |  <pre> graph TD subgraph Main_Function [Main Function] StartMain([Start main(void)]) --> InitMain[a = 100 b = 200 เป็น integer] InitMain --> PrintBeforeMain[/Before swap: value of a: %d ค่า a/] PrintBeforeMain --> PrintBeforeMainB[/Before swap: value of b: %d ค่า b/] PrintBeforeMainB --> CallSwap[swap(a,b)] CallSwap --> PrintAfterMainA[/After swap: value of a: %d ค่า a/] PrintAfterMainA --> PrintAfterMainB[/After swap: value of b: %d ค่า b/] PrintAfterMainB --> EndMain([End]) end subgraph Swap_Function [Swap Function] StartSwap([Start void swap (int asw, bsw)]) --> InitTemp[temp = asw] InitTemp --> SwapStep[asw = bsw bsw = temp] SwapStep --> EndSwap([End]) end </pre> |