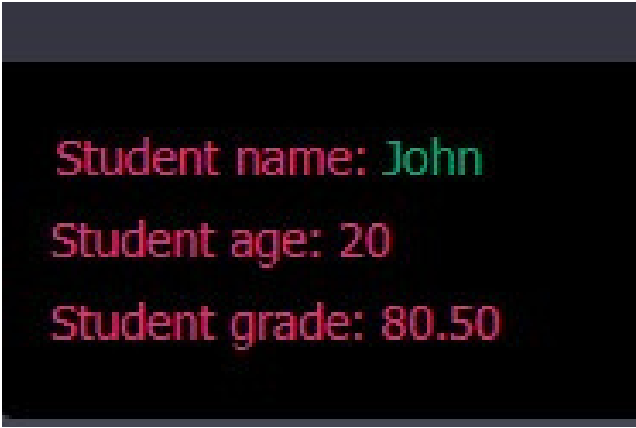
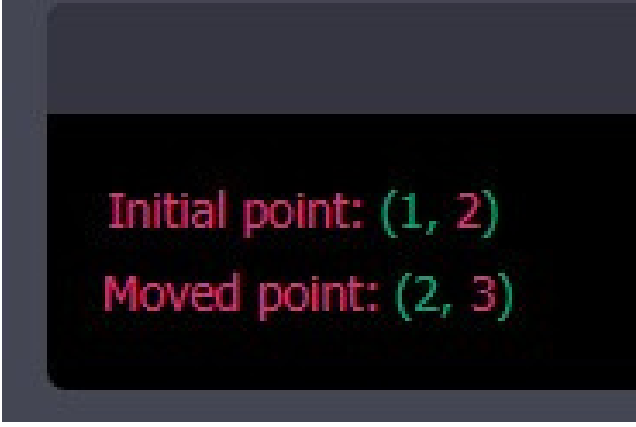
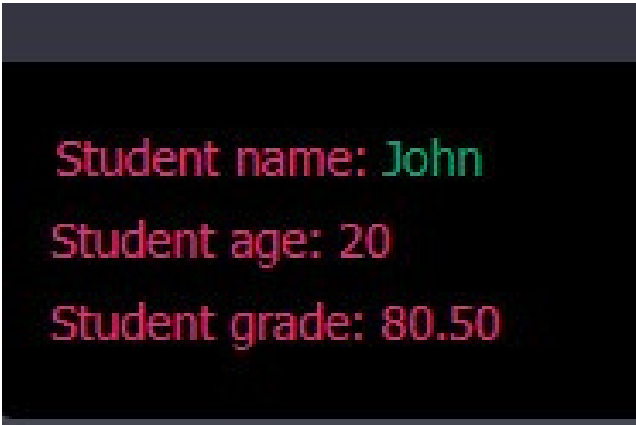


ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Structure อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Structure คือ โครงสร้างข้อมูลแบบหนึ่ง ที่รวมตัวแปรหลายๆ ตัวแปรไว้ด้วยกัน โดยที่ไม่จำเป็นต้องมีชนิดข้อมูล (Data type) เหมือนกัน เช่น</p> <pre>struct Laptop { //data_type     var_name     char *brand;     char *model;     int ram;     int ssd;     int price; };</pre>	<pre>#include &lt;stdio.h&gt; #include &lt;string.h&gt;  struct student {     char name[50];     int age;     float grade; };  int main(void) {     struct student s1;     strcpy(s1.name, "John");     s1.age = 20;     s1.grade = 80.5;     printf("Student name: %s\n", s1.name);     printf("Student age: %d\n", s1.age);     printf("Student grade: %.2f\n", s1.grade);      return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

ข้อที่ 2 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by reference และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>วิธีการส่งผ่านค่าตัวแปรประเภท struct แบบ pass by reference</p> <p>คือการใช้สัญลักษณ์ (*ตัวแปรใหม่).ตัวแปรในสตริง = ค่าต่างๆ หรือ ในรูปแบบ ตัวแปรใหม่-&gt;ตัวแปรในสตริง = ค่าต่างๆ เป็นต้น</p>	<pre>#include &lt;stdio.h&gt;  // Define a struct for a point struct point {     int x;     int y; };  // Function that takes a point by reference void move_point(struct point *p) {     // Modify the x and y values of the point     p-&gt;x = p-&gt;x + 1;     p-&gt;y = p-&gt;y + 1; }  int main(void) {     // Create a point     struct point p = {1, 2};      // Print the point's initial x and y values     printf("Initial point: (%d, %d)\n", p.x, p.y);      // Call the function to move the point     move_point(&amp;p);      // Print the point's x and y values after the function call     printf("Moved point: (%d, %d)\n", p.x, p.y);      return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
	

ข้อที่ 3 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by value และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>วิธีการส่งผ่านค่าตัวแปรประเภท struct แบบ pass by value คือการใช้สัญลักษณ์ ตัวแปรใหม่.ตัวแปรในสตรัค = ค่าต่างๆ เป็นต้น เช่น</p> <p>s1.grade = 80.5 ;</p>	<pre>#include &lt;stdio.h&gt; #include &lt;string.h&gt;  struct student {     char name[50];     int age;     float grade; };  int main(void) {     struct student s1;     strcpy(s1.name, "John");     s1.age = 20;     s1.grade = 80.5;     printf("Student name: %s\n", s1.name);     printf("Student age: %d\n", s1.age);     printf("Student grade: %.2f\n", s1.grade);      return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

## ข้อที่ 4 จงอธิบายการใช้ Struct ร่วมกับ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

## คำอธิบาย

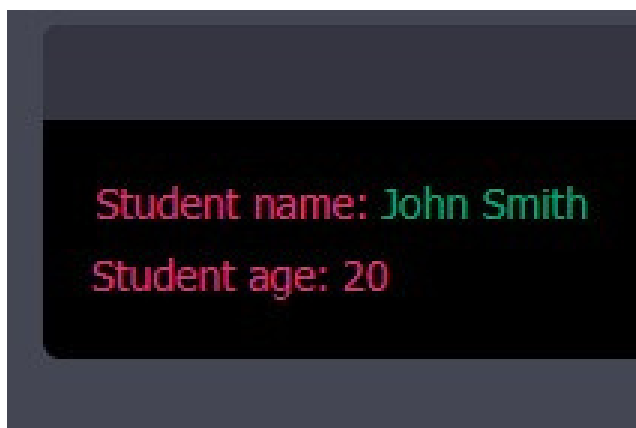
Structure ร่วมกับ pointer การทำงานจะคล้ายคลึงกับ pass by reference โดยการกำหนดชื่อใหม่ของstruct ให้เป็น pointer เพื่อให้สามารถ เก็บstruct ได้มากกว่า 1 นั่นเอง หรือนำไปทำเป็นฟังก์ชันรับค่าก็ได้เช่นกัน เช่น

```
struct student *s  
s = new struct student [2]
```

## Code ตัวอย่าง

```
#include <stdio.h>  
  
// Define a struct for a student  
struct student {  
    char *name;  
    int age;  
};  
  
// Function that takes a student by reference  
void print_student(struct student *s) {  
    printf("Student name: %s\n", s->name);  
    printf("Student age: %d\n", s->age);  
}  
  
int main(void) {  
    // Create a student  
    struct student s = {"John Smith", 20};  
  
    // Call the function to print the student's  
    information  
    print_student(&s);  
  
    return 0;  
}
```

## ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)



ข้อที่ 5 จงประยุกต์ใช้ Function ส่งผ่าน Struct แบบ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>คือการใช้ fuction ขึ้นมาโดยป้อนค่าตัวแปร สตรัคแบบ pointer ไปใน agument ของ function ทำให้สามารถรับค่าที่เป็นแบบ address เข้ามาทำงานในfunction ได้ โดยที่ค่าของadress ชื่อ ฟังก์ชันก็ยังคงเป็น address เดิมแต่มี value ที่เปลี่ยนไปได้ ดังตัวอย่าง</p>	<pre>#include &lt;stdio.h&gt;  // Define a struct for a point struct point {     int x;     int y; };  // Function that takes a point by reference void move_point(struct point *p) {     // Modify the x and y values of the point     p-&gt;x = p-&gt;x + 1;     p-&gt;y = p-&gt;y + 1; }  int main(void) {     // Create a point     struct point p = {1, 2};      // Print the point's initial x and y values     printf("Initial point: (%d, %d)\n", p.x, p.y);      // Call the function to move the point     move_point(&amp;p);      // Print the point's x and y values after the function call     printf("Moved point: (%d, %d)\n", p.x, p.y);      return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
