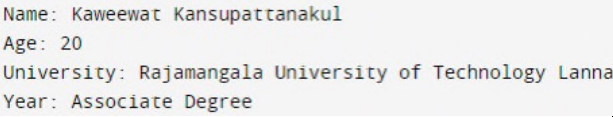
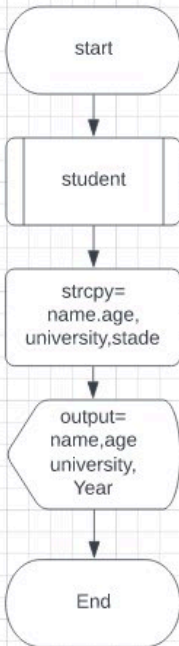
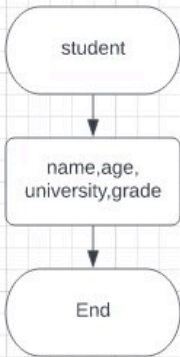
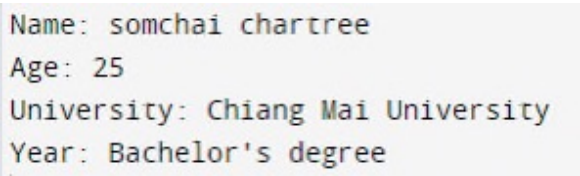


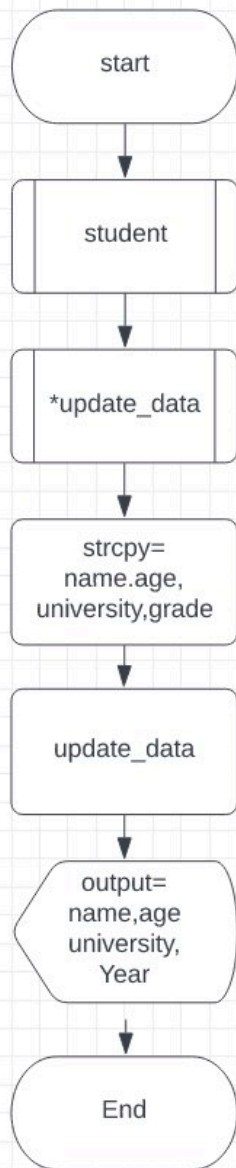
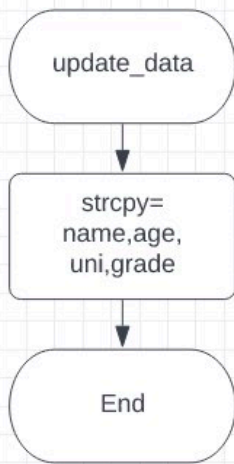
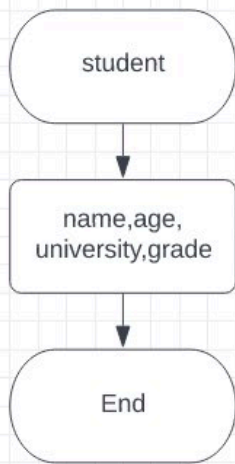
## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

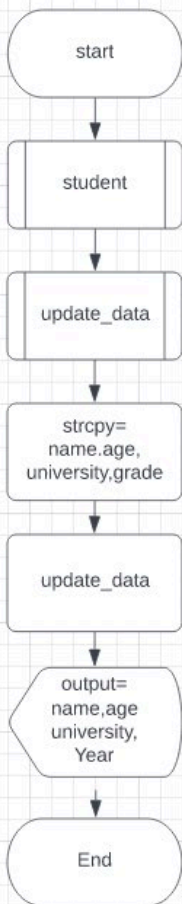
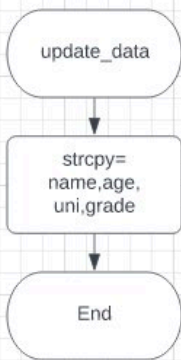
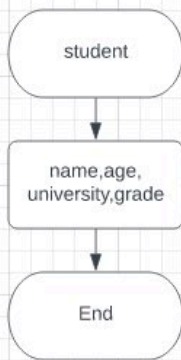
ข้อที่ 1 จงอธิบายความหมายของ Structure อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>ให้struct เก็บค่าของstudent แล้วนำไปแสดงผล</p> <p>ในตอนสุดท้าย โดน ตยย คือการแสดงชื่อ อายุ</p> <p>มหาวิทยาลัย และ ชั้นปี</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 //set value 4 struct student { 5     char name[50]; 6     int age; 7     char university[100]; 8     char grade[20]; 9 }; 10 11 int main() { 12     struct student stds; //set student to stds 13 14     //college data user 15     strcpy(stds.name, "Kaweewat Kansupattanakul"); 16     stds.age = 20; 17     strcpy(stds.university, "Rajamangala University of Technology Lanna"); 18     strcpy(stds.grade, "Associate Degree"); 19 20     //print zone 21     printf("Name: %s\n", stds.name); 22     printf("Age: %d\n", stds.age); 23     printf("University: %s\n", stds.university); 24     printf("Year: %s\n", stds.grade); 25 26     return 0; 27 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	



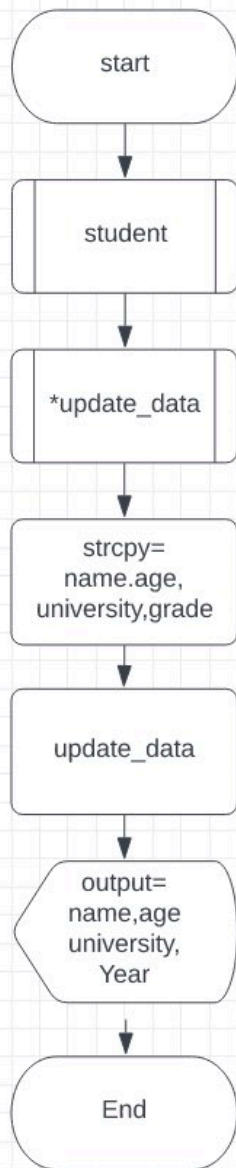
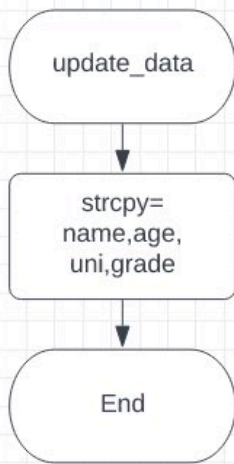
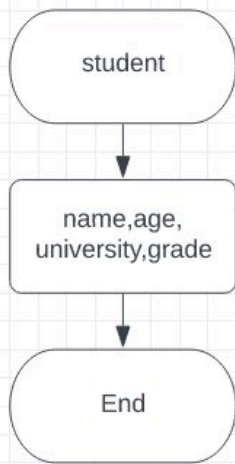
ข้อที่ 2 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by reference และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่างการใช้งาน
<p>ให้structรับค่า ตัวแปร และใช้update เป็นpass by ref ในการเปลี่ยนค่าตัวแปร</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct student { 4     char name[50]; 5     int age; 6     char uni[100]; 7     char grade[20]; 8 }; 9 //update user name filed 10 void update_data(struct student *stds) { 11     strcpy(stds-&gt;name, "somchai chartree"); 12     stds-&gt;age = 25; 13     strcpy(stds-&gt;uni, "Chiang Mai University"); 14     strcpy(stds-&gt;grade, "Bachelor's degree"); 15 } 16 17 int main() { 18     struct student stds; 19 20     // main filed struct 21     strcpy(stds.name, "Kaweewt Kansupattanakul"); 22     stds.age = 20; 23     strcpy(stds.uni, "Rajamangala University of Technology Lanna"); 24     strcpy(stds.grade, "Associate Degree"); 25 26     //using pass by ref 27     update_data(&amp;stds); 28 29     // Print the updated fields of the structure 30     printf("Name: %s\n", stds.name); 31     printf("Age: %d\n", stds.age); 32     printf("University: %s\n", stds.uni); 33     printf("Year: %s\n", stds.grade); 34 35     return 0; 36 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	



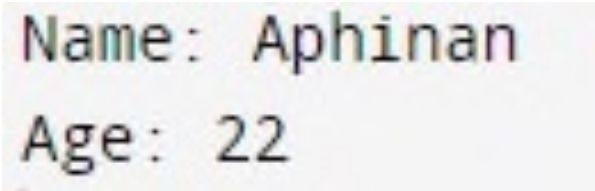
ข้อที่ 3 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by value และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct student { 4     char name[50]; 5     int age; 6     char uni[100]; 7     char grade[20]; 8 }; 9 //update user name filed 10 void update_data(struct student *stds) { 11     strcpy(stds -&gt; name, "somchai chartree"); 12     stds-&gt;age = 25; 13     strcpy(stds -&gt; uni, "Chiang Mai University"); 14     strcpy(stds -&gt; grade, "Bachelor's degree"); 15 } 16 17 int main() { 18     struct student stds; 19 20     // main filed struct 21     strcpy(stds.name, "Kaweewt Kansupattanakul"); 22     stds.age = 20; 23     strcpy(stds.uni, "Rajamangala University of Technology Lanna"); 24     strcpy(stds.grade, "Associate Degree"); 25 26 27     // Print the updated fields of the structure 28     printf("Name: %s\n", stds.name); 29     printf("Age: %d\n", stds.age); 30     printf("University: %s\n", stds.uni); 31     printf("Year: %s\n", stds.grade); 32 33     return 0; 34 </pre>
<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct student { 4     char name[50]; 5     int age; 6     char uni[100]; 7     char grade[20]; 8 }; 9 //update user name filed 10 void update_data(struct student *stds) { 11     strcpy(stds -&gt; name, "somchai chartree"); 12     stds-&gt;age = 25; 13     strcpy(stds -&gt; uni, "Chiang Mai University"); 14     strcpy(stds -&gt; grade, "Bachelor's degree"); 15 } 16 17 int main() { 18     struct student stds; 19 20     // main filed struct 21     strcpy(stds.name, "Kaweewt Kansupattanakul"); 22     stds.age = 20; 23     strcpy(stds.uni, "Rajamangala University of Technology Lanna"); 24     strcpy(stds.grade, "Associate Degree"); 25 26 27     // Print the updated fields of the structure 28     printf("Name: %s\n", stds.name); 29     printf("Age: %d\n", stds.age); 30     printf("University: %s\n", stds.uni); 31     printf("Year: %s\n", stds.grade); 32 33     return 0; 34 </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>



ข้อที่ 4 จงอธิบายการใช้ Struct ร่วมกับ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>การแสดงผลออกมาเพื่อเรียกใช้งานผ่านpointer</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 // Define a struct type called "Person" 4 struct Student { 5     char name[50]; 6     int age; 7     char uni[100]; 8     char grade[20]; 9 }; 10 11 // Declare a function that takes a pointer to a "Person" struct as an argument 12 void printStudent(struct Student *stds) { 13     printf("Name: %s\n", stds-&gt;name); 14     printf("Age: %d\n", stds-&gt;age); 15     printf("University: %s\n", stds-&gt;uni); 16     printf("Year: %s\n", stds-&gt;grade); 17 } 18 19 int main() { 20     // struct student 21     struct Student stds = {"Kaweewat Kansupattanakul" , 22                             20, 23                             "Rajamangala University of Technology Lanna", 24                             "Associate Degree" }; 25 26     // Call the function and pass the address of the struct as an argument 27     printStudent(&amp;stds); 28 29     return 0; 30 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>Name: Kaweewat Kansupattanakul  Age: 20  University: Rajamangala University of Technology Lanna  Year: Associate Degree</p>	





ข้อที่ 5 จงประยุกต์ใช้ Function ส่งผ่าน Struct แบบ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>ใช้func การการแสดงชื่อ และใช้pointer ในการประกาศint เพื่อเข้ามาเช็คอายุและแสดงชื่อคนที่อายุตามที่ตั้งไว้ออกมา</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 // call student for function 4 typedef char* (*Student)(int); 5 6 // struct student to find student data and int 7 struct StudentData { 8     Student stds; 9     int age; 10 }; 11 12 // fine struct studentData 13 void printStudent(struct StudentData data) { 14 15     // Call the function stored in the "p" field of the struct 16     char* name = data.stds(data.age); 17     printf("Name: %s\n", name); 18     printf("Age: %d\n", data.age); 19     return 0; 20 } 21 22 // return char name 23 char* getName(int age) { 24     if (age &lt; 20) { 25         return "Kaweewat"; 26     } else { 27         return "Aphinan"; 28     } 29 } 30 31 int main() { 32     // use function struct and getname 33     struct StudentData data = {&amp;getName, 22}; 34 35     // call data 36     printStudent(data); 37 38     return 0; 39 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	

```

1 #include <stdio.h>
2
3 // call student for function
4 typedef char* (*Student)(int);
5
6 // struct student to find student data and int
7 struct StudentData {
8     Student stds;
9     int age;
10 };
11
12 // fine struct studentData
13 void printStudent(struct StudentData data) {
14
15     // Call the function stored in the "p" field of the struct
16     char* name = data.stds(data.age);
17     printf("Name: %s\n", name);
18     printf("Age: %d\n", data.age);
19     return 0;
20 }
21
22 // return char name
23 char* getName(int age) {
24     if (age < 20) {
25         return "Kaweewat";
26     } else {
27         return "Aphinan";
28     }
29 }
30
31 int main() {
32     // use function struct and getname
33     struct StudentData data = {&getName, 22};
34
35     // call data
36     printStudent(data);
37
38     return 0;
39 }

```