

ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวัดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ Structure อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>โครงสร้างข้อมูลประเภทตัวแปร</p> <p>ข้อมูลที่เก็บ</p> <p>1) ค่าชี้ที่ไปยัง พล็อก หน้า หน้าจอที่ต้องการ นำแสดงผลในหน้าจอ</p> <p>2) ตชุดของตัวแปรที่ต้องการ เช่น ตัวแปร เวลา และ ตัวแปรที่ไม่ใช่ตัวแปร</p> <p>3) โครงสร้าง คือ เก็บชุดข้อมูล ที่มีความซับซ้อน กันในโปรแกรม</p>	<pre># include <stdio.h> int main () { printf ("Hello world\n"); return 0 ; }</pre> <pre>1 #include <stdio.h> 2 3▼ int main() { 4 printf ("Hello world\n"); 5 return 0 ; 6 }</pre>
ผลลัพธ์ของ Code (Captureภาพแม่รูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD start((start)) --> int[int main] int --> hello[Hello World] hello --> end((End)) </pre>

ข้อที่ 2 จะแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by reference และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>กิจกรรมโครงสร้างที่ใช้กัน “บุคคล” หมายความว่า “คน” หากพิมพ์ภาษา C ให้ใช้ printf - person ก็จะได้รับชื่อ โครงสร้าง บุคคล ที่มีชื่อเป็นภาษาไทย ให้ตั้งค่าใน struct ของตัวแปร person ที่มีชื่อ เบื้องต้นของตัวแปร</p>	<pre> 1 #include <stdio.h> 2 3 struct person { 4 char name[20]; 5 int age; 6 }; 7 8 void print_person(struct person *p) { 9 printf ("Name : %s , Age : %d\n" , p->name, p->age); 10 } 11 12 int main() { 13 struct person p= {"pim",19}; 14 print_person(&p); 15 return 0 ; 16 }</pre>
ผลลัพธ์ของ Code (Captureภาพแม่รูป)	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD StartMain([start main]) --> PrintPersonVoid[void print_person (&struct person p)] PrintPersonVoid --> LocalP[struct person p = {"pim", 19}] LocalP --> PrintPersonFunc[print_person (&p)] PrintPersonFunc --> EndMain([end]) EndMain --> EndPrintPersonFunc([end]) </pre>

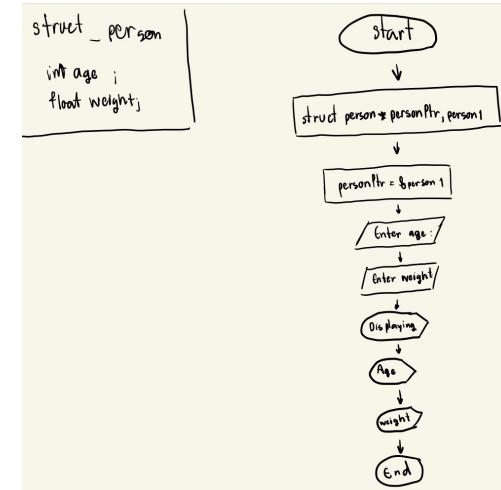
ข้อที่ 3 จะแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by value และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>ในฟังชัน printPoint ถือว่ารับ point เป็นพารามิเตอร์ แล้ว print pointer ออกมา ต้องมี การcopy ค่าทุกส่วนมาเป็นตัวของตัวเอง ไม่ว่าจะเป็นค่า x,y หรือค่า struct Point ที่เก็บค่า x,y ไว้</p> <p>ในส่วนของ point คงต้องรับในรูปแบบ struct Point ให้ได้</p> <p>เดินทางไปที่จุดที่ต้องการ</p>	<pre> 1 #include <stdio.h> 2 3 ▶ typedef struct { 4 int x; 5 int y; 6 }Point; 7 8 ▶ void printPoint(Point p){ 9 printf("(%, %)\n", p.x, p.y); 10 } 11 ▶ int main(){ 12 13 Point p = {1,2}; 14 15 printPoint(p); 16 17 return 0; 18 }</pre>
ผลลัพธ์ของ Code (Captureพื้นจอ)	Flow chart ของ Code ตัวอย่าง

ข้อที่ 4 จะอธิบายการใช้ Struct ร่วมกับ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer

คำอธิบาย	Code ตัวอย่าง
<p>โครงสร้าง struct คือ กลุ่มข้อมูล ไม่ต้องเก็บข้อมูลในหนึ่งที่ต้องการ หนึ่งเดียว ต้อง person 1 ก็ต้อง person 2 แต่ต้องการ person 1 ไม่ต้อง person 2 โดย personptr = &person1 จะมีชื่อ person1 ในหนึ่ง personptr personptr -> age หรือใน (*personptr).age personptr -> weight หรือใน (*personptr).weight</p>	<pre>> ... 1 #include <stdio.h> 2 ▼ struct person{ 3 int age; 4 float weight; 5 }; 6 7 ▼ int main(){ 8 9 struct person *personPtr, person1; 10 personPtr = &person1; 11 printf("Enter age :"); 12 scanf("%d", &personPtr->age); 13 printf("Enter weight :"); 14 scanf("%f", &personPtr->weight); 15 16 printf("Displaying:\n"); 17 printf("Age: %d\n", personPtr->age); 18 printf("Weight: %f", personPtr->weight); 19 return 0; 20 }</pre>
ผลลัพธ์ของ Code (Captureภาพรูป)	Flow chart ของ Code ตัวอย่าง

```
> make -s
> ./main
Enter age :98
Enter weight :56
Displaying:
Age: 98
Weight: 56.000000
```



ข้อที่ 5 จะประยุกต์ใช้ Function ส่งผ่าน Struct แบบ Pointer ยกตัวอย่างการใช้งาน พิจารณา Code ตัวอย่างการใช้งาน Pointer

คำอธิบาย	Code ตัวอย่าง
<p>กำหนด struct ชื่อ example_struct โครงสร้างมี 3 อย่าง a, b, c กำหนดฟังก์ชัน ที่รับมาเป็นค่า struct ที่ต้องการ example_struct เป็นตัวถูกเพนท์โดย จะนำค่าที่ต้องการส่ง ไปเรียกใช้ในฟังก์ชัน ที่รับมาเป็นค่า struct pointer</p>	<pre> 1 #include <stdio.h> 2 3 struct example_struct{ 4 int a; 5 float b; 6 char c; 7 }; 8 9 void function(struct example_struct* example){ 10 printf("a = %d\n", example->a); 11 printf("b = %f\n", example->b); 12 printf("c = %c\n", example->c); 13 } 14 int main(){ 15 16 struct example_struct example = {1, 2.5, 'c'}; 17 function(&example); 18 19 return 0; 20 }</pre>
ผลลัพธ์ของ Code (Captureภาพแม่รูป)	Flow chart ของ Code ตัวอย่าง
<pre> > make -s > ./main a = 1 b = 2.500000 c = c ></pre>	<pre> graph TD start_main([start main]) --> void_func[void function(struct example_struct* example)] void_func --> printf_a["printf(\"a = %d\\n\", example->a)"] void_func --> printf_b["printf(\"b = %f\\n\", example->b)"] void_func --> printf_c["printf(\"c = %c\\n\", example->c)"] printf_a --> end([end]) printf_b --> end printf_c --> end </pre>