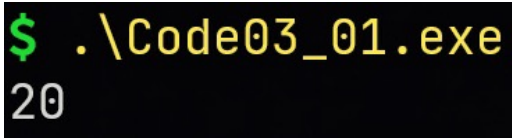


## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

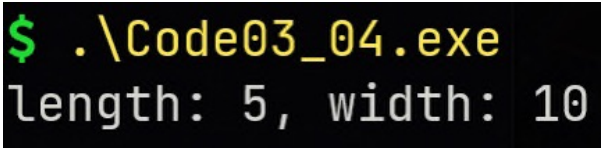
จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

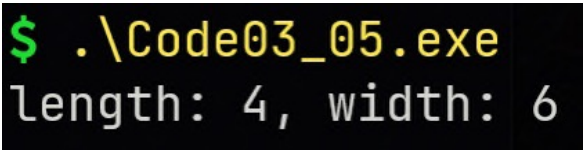
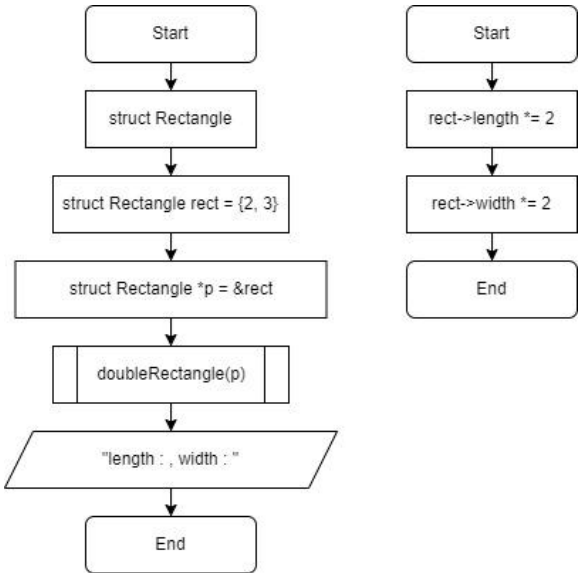
ข้อที่ 1 จงอธิบายความหมายของ Structure อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน

คำอธิบาย	Code ตัวอย่าง
<p>Structure คือการเก็บข้อมูลในชุดใดก็ได้ ซึ่งประกอบด้วย                  โครงสร้างที่ Structure นี้ สามารถเก็บข้อมูล                  ใดก็ได้ (.) ของ ตามชื่อของ Structure StructName . Value</p>	<pre> * include &lt;stdio.h&gt; Struct student {     char name[20];     int age;     char sex;     float gpa; }; typedef struct student stds; int main () {     stds aboy;     aboy.age = 10;     printf("%d", age.aboy);     return 0; }                 </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD     Start([Start]) --&gt; Init[student student(stds)]     Init --&gt; Decl[stds aboy]     Decl --&gt; Assign[aboy.age = 20]     Assign --&gt; Print[/show (aboy.age)/]     Print --&gt; Stop([Stop])                 </pre>

ข้อที่ 2 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by reference และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การส่งค่าตัวแปรประเภท Struct แบบ Pass by reference นั้น จะทำการเปลี่ยนพจน์ของ Argument ที่ส่ง an</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct Rectangle { 4     int length; 5     int width; 6 }; 7 8 void doubleRectangleByReference(struct Rectangle *rect) { 9     rect-&gt;length *= 2; 10    rect-&gt;width *= 2; 11    printf("In Function -&gt; length: %d, width: %d\n", rect-&gt;length, rect-&gt;width); 12 } 13 14 int main() { 15     struct Rectangle rect = {2, 3}; 16     doubleRectangleByReference(&amp;rect); 17     printf("Out Function -&gt; length: %d, width: %d\n", rect.length, rect.width); 18     return 0; 19 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<pre> \$ .\Code03_02.exe In Function -&gt; length: 4, width: 6 Out Function -&gt; length: 4, width: 6 </pre>	<pre> graph TD     subgraph Left_Flowchart         Start1[Start] --&gt; struct1[struct Rectangle]         struct1 --&gt; rect_init[struct Rectangle rect = {2, 3}]         rect_init --&gt; func_call[doubleRectangleByReference(&amp;rect)]         func_call --&gt; output[/Out Function -&gt; length : , width :/]         output --&gt; End1[End]     end     subgraph Right_Flowchart         Start2[Start] --&gt; len_mult[rect-&gt;length *= 2]         len_mult --&gt; width_mult[rect-&gt;width *= 2]         width_mult --&gt; End2[End]     end </pre>

ข้อที่ 3 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท Struct แบบ Pass by value และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การส่งค่าตัวแปรประเภท Struct แบบ Pass by Value นั้น จะทำการเปลี่ยนแปลงข้อมูล Argument ที่ส่งไป ทำจนไปถึงข้อมูลนอก Function เมื่อ Return ตัวแปรกลับ</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct Rectangle { 4     int length; 5     int width; 6 }; 7 8 void doubleRectangleByValue(struct Rectangle rect) { 9     rect.length *= 2; 10    rect.width *= 2; 11    printf("In Function -&gt; length: %d, width: %d\n", rect.length, rect.width); 12 } 13 14 int main() { 15     struct Rectangle rect = {2, 3}; 16     doubleRectangleByValue(rect); 17     printf("Out Function -&gt; length: %d, width: %d\n", rect.length, rect.width); 18     return 0; 19 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<pre> \$ .\Code03_03.exe In Function -&gt; length: 4, width: 6 Out Function -&gt; length: 2, width: 3 </pre>	<pre> graph TD     subgraph Left_Path [ ]         direction TB         L_Start([Start]) --&gt; L_Struct[struct Rectangle]         L_Struct --&gt; L_Rect["struct Rectangle rect = {2, 3}"]         L_Rect --&gt; L_Func[doubleRectangleByValue(rect)]         L_Func --&gt; L_Out[/Out Function -&gt; length : , width :/]         L_Out --&gt; L_End([End])     end     subgraph Right_Path [ ]         direction TB         R_Start([Start]) --&gt; R_Length["rect.length *= 2"]         R_Length --&gt; R_Width["rect.width *= 2"]         R_Width --&gt; R_End([End])     end </pre>

ข้อที่ 4 จงอธิบายการใช้ Struct ร่วมกับ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>ให้ *p เป็นตัวชี้ของ struct ที่ประกาศไว้</p> <p>ตัวท่อนค่า struct ใดโดยให้ *p</p> <p>จะพบค่าของ struct ที่ประกาศไว้เป็นของค่า</p> <p>*p</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct Rectangle { 4     int length; 5     int width; 6 }; 7 8 int main() { 9     struct Rectangle rect = {2, 3}; 10    struct Rectangle *p = &amp;rect; 11    p-&gt;length = 5; 12    p-&gt;width = 10; 13    printf("length: %d, width: %d\n", rect.length, rect.width); 14    return 0; 15 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD     Start([Start]) --&gt; Decl[struct Rectangle]     Decl --&gt; Init[struct Rectangle *p = &amp;rect]     Init --&gt; RectDef[struct Rectangle rect = {2, 3}]     RectDef --&gt; Len[p-&gt;length = 5]     Len --&gt; Wid[p-&gt;width = 10]     Wid --&gt; Print[/"length :", width :"/]     Print --&gt; End([End])     </pre>

ข้อที่ 5 จงประยุกต์ใช้ Function ส่งผ่าน Struct แบบ Pointer ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>ให้ *p เก็บข้อมูล struct ที่ประกาศไว้</p> <p>พ้องกับค่า struct ในลิสต์ *p</p> <p>แล้วให้ p เข้าไปใน Function</p> <p>จ.พบว่า Struct ที่ประกาศไว้มีการเปลี่ยนแปลง</p> <p>ตาม Function ที่เราได้ใส่ *p เข้าไปใน</p>	<pre> 1 #include &lt;stdio.h&gt; 2 3 struct Rectangle { 4     int length; 5     int width; 6 }; 7 8 void doubleRectangle(struct Rectangle *rect) { 9     rect-&gt;length *= 2; 10    rect-&gt;width *= 2; 11 } 12 13 int main() { 14     struct Rectangle rect = {2, 3}; 15     struct Rectangle *p = &amp;rect; 16 17     doubleRectangle(p); 18     printf("length: %d, width: %d\n", rect.length, rect.width); 19     return 0; 20 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD     subgraph Left_Flow [ ]         direction TB         L_Start([Start]) --&gt; L_Struct[struct Rectangle]         L_Struct --&gt; L_Rect["struct Rectangle rect = {2, 3}"]         L_Rect --&gt; L_Ptr["struct Rectangle *p = &amp;rect"]         L_Ptr --&gt; L_Func[doubleRectangle(p)]         L_Func --&gt; L_Print[/length : , width : /]         L_Print --&gt; L_End([End])     end     subgraph Right_Flow [ ]         direction TB         R_Start([Start]) --&gt; R_Length["rect-&gt;length *= 2"]         R_Length --&gt; R_Width["rect-&gt;width *= 2"]         R_Width --&gt; R_End([End])     end </pre>