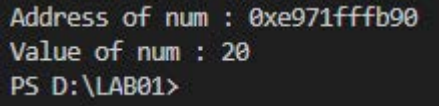
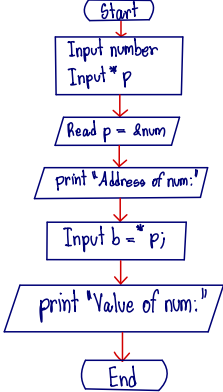
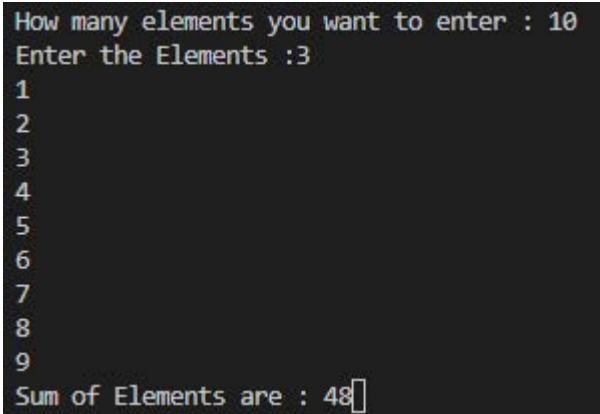
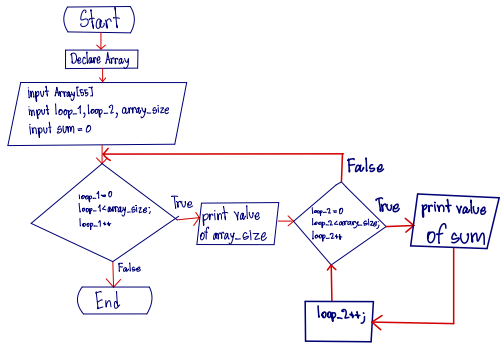
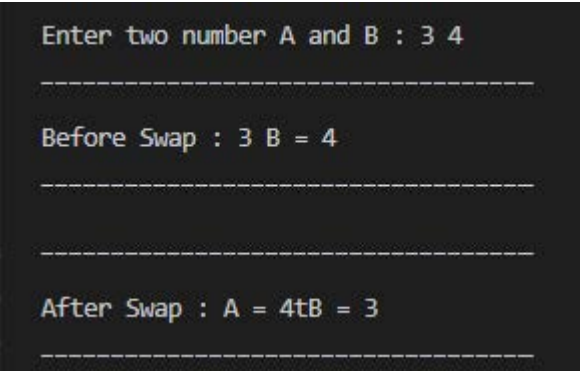
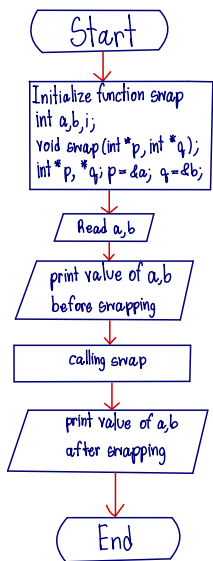


## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

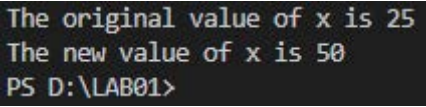
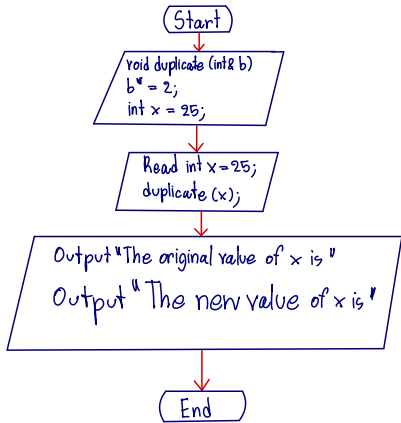
จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

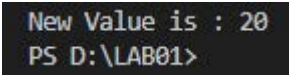
คำอธิบาย	Code ตัวอย่าง
<p>พอยน์เตอร์ (Pointer) คือ ตัวแปรที่เก็บค่าของที่อยู่ของตัวแปร การให้พอยน์เตอร์จะทำให้เราสามารถเข้าถึงข้อมูลได้โดยไม่ต้องผ่านหน่วยความจำ แทนที่เราจะใส่ชื่อตัวแปร พอยน์เตอร์นี้จะใช้สำหรับ low-level programming เช่น การเขียนโปรแกรมในภาษา C หลักการทำงานของพอยน์เตอร์จะมีความคล้ายคลึงกับตัวแปรทั่วไป</p>	<pre> 1  #include &lt;iostream&gt; 2  using namespace std; 3 4  int main() 5  { 6      int num = 20; 7      int * p; 8      p = &amp;num; 9 10     cout &lt;&lt; "Address of num : " &lt;&lt; p &lt;&lt; "\n"; 11 12     int b = *p; 13     cout &lt;&lt; "Value of num : " &lt;&lt; *p &lt;&lt; "\n"; 14 15     return 0; 16 }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD     Start([Start]) --&gt; Input1[Input number Input * p]     Input1 --&gt; Read[Read p = &amp;num]     Read --&gt; Print1[/print "Address of num:"/]     Print1 --&gt; Input2[Input b = *p;]     Input2 --&gt; Print2[/print "Value of num:"/]     Print2 --&gt; End([End]) </pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>เปลี่ยนตำแหน่งการอ้างอิงของตัวชี้ให้ชี้ไปยัง Function ที่ส่งกลับค่า Pointer ในฟังก์ชันอื่นได้ และเพื่อใช้ในการนำข้อมูลที่ Pointer ที่ส่งกลับมาจากฟังก์ชันใน Function ได้</p>	<pre> LAB01-2.cpp &gt; ... 1  #include &lt;iostream&gt; 2  #include &lt;conio.h&gt; 3  using namespace std; 4 5  int main() 6  { 7      int Array[55]; 8      int loop_1, loop_2, array_size; 9      int sum=0; 10     cout&lt;&lt;"How many elements you want to enter : "; 11     cin&gt;&gt;array_size; 12     cout&lt;&lt;"Enter the Elements : "; 13     for(loop_1=0;loop_1&lt;array_size;loop_1++) 14     { 15         cin&gt;&gt;Array[loop_1]; 16     } 17     cout&lt;&lt;"Sum of Elements are : "; 18 19     for(loop_2=0;loop_2&lt;array_size;loop_2++) 20     { 21         sum=sum+Array[loop_2]; 22     } 23     cout&lt;&lt;sum; 24     getch(); 25 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
 <pre> How many elements you want to enter : 10 Enter the Elements : 3 1 2 3 4 5 6 7 8 9 Sum of Elements are : 48 </pre>	 <pre> graph TD     Start([Start]) --&gt; DeclareArray[Declare Array]     DeclareArray --&gt; Input[Input Array[55] Input loop_1, loop_2, array_size Input sum = 0]     Input --&gt; Loop1Start{loop_1 = 0 loop_1 &lt; array_size loop_1++}     Loop1Start -- True --&gt; PrintArraySize[print value of array_size]     PrintArraySize --&gt; Loop2Start{loop_2 = 0 loop_2 &lt; array_size loop_2++}     Loop2Start -- True --&gt; PrintSum[print value of sum]     PrintSum --&gt; Loop2Start     Loop2Start -- False --&gt; Loop1Start     Loop1Start -- False --&gt; End([End]) </pre>

ข้อที่ 3 จงอธิบายเรื่อง Pointer Functionยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>เป็นฟังก์ชันที่มีโครงสร้างเหมือนฟังก์ชันอื่นแต่ไปใส่พารามิเตอร์ชื่อฟังก์ชันเป็นอาร์กิวเมนต์โดยตัวชี้ตัวแปร          ตัวอย่าง เช่น <code>typename (*functionname)(parameter);</code> การเขียนพอยน์เตอร์ฟังก์ชันมีในภาษา C          มาก่อนที่จะมีการเขียนโปรแกรมเชิงวัตถุ ซึ่งใช้รับแบบการเขียนสลับรอมองเป็นวัตถุ และใช้ไปจนทุก          ชั่วโมงจำนวนได้ที่เขียน</p>	<pre> G:\Lab01-3app &gt; @ menu 1  #include &lt;stdio.h&gt; 2  #include &lt;conio.h&gt; 3 4  main() 5  { 6      int a,b,i; 7      void swap(int *p,int *q); 8      int *p, *q; p = &amp;a; q = &amp;b; 9 10     printf("Enter two number A and B : "); 11     scanf("%d%d",&amp;a,&amp;b); 12 13     for(i=0;i&lt;35;i++) 14         printf("_");printf("\n"); 15 16     printf("Before Swap : %d\tB = %d\n",a,b); 17 18     for(i=0;i&lt;35;i++) printf("_"); 19     printf("\n"); 20 21     for(i=0;i&lt;35;i++) 22         printf("_");printf("\n"); 23 24     swap(p,q); for(i=0;i&lt;35;i++) 25         printf("_");printf("\n"); 26 27     getch(); 28     return 0; 29 } 30 31 /* Swap Operations */ 32 33 void swap(int *p, int *q) 34 { 35     int temp = 0; temp = *p; 36     *p = *q; 37     *q = temp; 38     printf("After Swap : A = %d\tB = %d\n",*p,*q); 39 }         </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre> graph TD     Start([Start]) --&gt; Init[Initialize function swap int a,b,i; void swap(int *p, int *q); int *p, *q; p = &amp;a; q = &amp;b;]     Init --&gt; Read[/Read a,b/]     Read --&gt; PrintBefore[/print value of a,b before swapping/]     PrintBefore --&gt; Call[calling swap]     Call --&gt; PrintAfter[/print value of a,b after swapping/]     PrintAfter --&gt; End([End])         </pre>

ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>ตัวแปรอาร์เรย์แบบไดนามิก เป็นตัวแปรอาร์เรย์ที่ไว้สำหรับเก็บข้อมูลที่แปรค่าไม่ทราบจำนวนที่แน่นอน หรือทราบแต่ต้องการให้ตัวแปรแบบไดนามิก เพื่อปรับขนาดการระบบ โดยที่ตัวแปรแบบอาร์เรย์จะปรับขนาดจำนวนสมาชิกให้เท่ากับจำนวนตัวแปรที่ต้องการใช้ในขณะนั้น ๆ นั่นคือเราสามารถปรับขนาดของระบบเมื่อต้องการเท่านั้น</p>	<pre> G: LAB01-4.cpp &gt; main() 4  int main() 5  { 6      int x, n; 7      cout &lt;&lt; "Enter the number of items : " &lt;&lt; endl; 8      cin &gt;&gt; n; 9      int *arr = new int(n); 10     cout &lt;&lt; "Enter " &lt;&lt; n &lt;&lt; " items" &lt;&lt; endl; 11     for (x = 0; x &lt; n; x++) 12     { 13         cin &gt;&gt; arr[x]; 14     } 15     cout &lt;&lt; "You entered : "; 16     for (x = 0; x &lt; n; x++) 17     { 18         cout &lt;&lt; arr[x] &lt;&lt; " "; 19     } 20     return 0; 21 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD     Start([Start]) --&gt; Input[Input value x, n Input *arr = new int(n);]     Input --&gt; Decision{x = 0; x &lt; n;}     Decision -- Next --&gt; Output[/Output *Enter the number of items: /]     Output --&gt; InputVal[/Input value/]     InputVal --&gt; Decision     Decision -- Done --&gt; End([End]) </pre>

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การส่งค่าไปยังฟังก์ชันที่เรียกใช้โดยส่งเป็นค่าตัวแปรที่อยู่ (Address) ของตัวแปรที่ไปซึ่งหากภายในฟังก์ชันมีการเปลี่ยนแปลงค่าของตัวแปรนั้นก็จะไป ก็จะมีผลทำให้ค่าของตัวแปรนั้นในโปรแกรมที่เรียกใช้เปลี่ยนไปด้วย</p>	<pre> LAB01-5.cpp X LAB01-5.cpp &gt; main() 1  #include &lt;iostream&gt; 2  using namespace std; 3 4  void duplicate(int&amp; b) 5  { 6      b*=2; 7  } 8 9  int main() 10 { 11     int x = 25; 12     cout &lt;&lt; "The original value of x is " &lt;&lt; x &lt;&lt; endl; 13     duplicate(x); 14     cout &lt;&lt; "The new value of x is " &lt;&lt; x; 15     return 0; 16 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
 <pre> The original value of x is 25 The new value of x is 50 PS D:\LAB01&gt; </pre>	 <pre> graph TD     Start([Start]) --&gt; Duplicate[void duplicate (int&amp; b) b*=2; int x = 25;]     Duplicate --&gt; Read[Read int x=25; duplicate(x);]     Read --&gt; Output[/Output "The original value of x is" Output "The new value of x is"/]     Output --&gt; End([End]) </pre>

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การส่งค่า (value) เป็น argument ของฟังก์ชัน ดังนั้นค่าที่ส่งให้ฟังก์ชันจริงไม่ส่งผลกระทบต่อตัวแปรของฟังก์ชัน</p>	<pre> LAB01-6.cpp X LAB01-6.cpp &gt; ... 1  #include &lt;stdio.h&gt; 2 3  int find_NewValue(int New_Value); 4 5  int main() 6  { 7      int n_value = 10; 8 9      int New_Value = find_NewValue(n_value); 10     printf("New Value is : %d\n",New_Value); 11 12     return 0; 13 } 14 15 int find_NewValue(int New_Value) 16 { 17     return New_Value + 10; 18 } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> graph TD     Start([Start]) --&gt; Input[/input value, new value/]     Input --&gt; Read[/Read New_Value + 10;/]     Read --&gt; Output[/Output Sum of New Value/]     Output --&gt; End([End]) </pre>