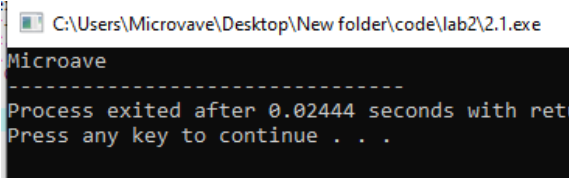
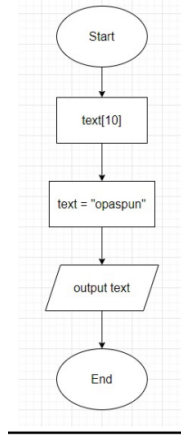
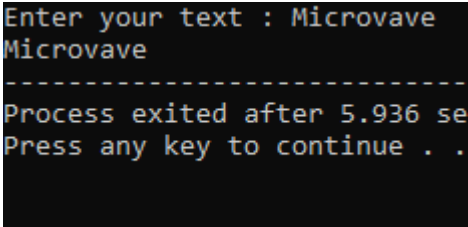
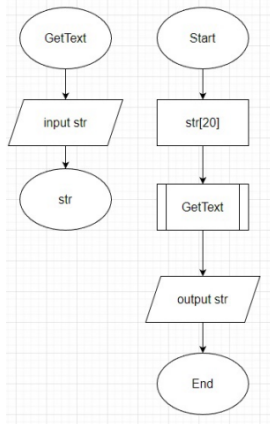
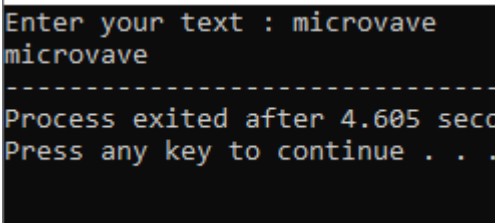
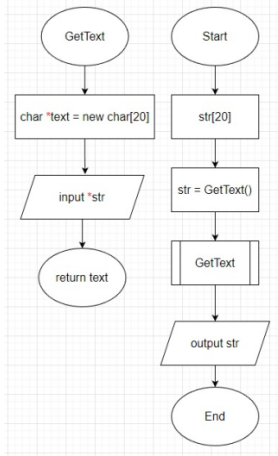


## ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

ข้อที่ 1 จงอธิบายความหมายของ String อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>String คือ "ข้อความ" มีลักษณะเป็น array ของตัวอักษรหรือเป็น array ของ char ซึ่ง char จะเก็บตัวอักษรได้เพียงตัวเดียว แต่ถ้าเป็น String จะเก็บตัวอักษรได้หลายตัว</p> <p>ดังนั้นเราสามารถส่งข้อมูลแบบ Pass by reference ได้</p>	<pre>#include &lt;stdio.h&gt; #include &lt;string.h&gt;  int main() {      char text[10];     strcpy(text, "Microave");     printf("%s", text);     return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre>graph TD     Start([Start]) --&gt; TextArray[text[10]]     TextArray --&gt; TextAssign[text = "opaspun"]     TextAssign --&gt; Output[/output text/]     Output --&gt; End([End])</pre>

ข้อที่ 2 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท String แบบ Pass by reference และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>เราสามารถตัวแปรแบบ "String" แล้วค่อยเรียกใช้ void function จากนั้น ทั้งการส่งและการรับค่าจะเกิดขึ้น Argument เช่น ใน void funtion อาจจะใช้ คำสั่ง gets เพื่อเก็บข้อมูลจากผู้ใช้ แล้วส่งคืนกลับไปหา main ผ่าน Argument โดย Argument ของ void funtion จะต้องใส่ pointer ด้วย</p>	<pre>#include &lt;stdio.h&gt;  void GetText( char *text );  int main(){     char str[20];     GetText(str);     printf("%s", str);     return 0; }  void GetText( char *text ) {     printf("Enter your text : ");     gets(text); }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

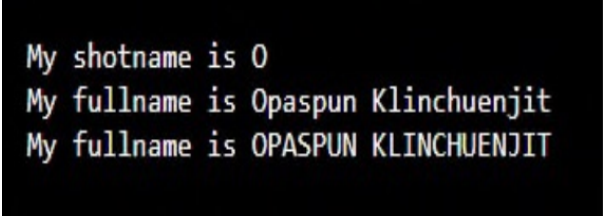
ข้อที่ 3 จงแสดงวิธีการส่งผ่านค่าตัวแปรประเภท String แบบ Pass by value และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>เราต้องสร้างตัวแปร String ขึ้นมาก่อน จากนั้นใช้ Function ที่มีการ return ค่ากลับ ใน Function</p> <p>ย่อนั้นอาจจะมีการใช้ตัวแปร char pointer เพื่อให้ทำการเก็บข้อมูลหรือชี้ตัวแปร เมื่อจบ Function ย่อยก็จะ return ค่ากลับมายา main</p> <p>ซึ่งค่าที่ return ค่ากลับมายา main ซึ่งค่าที่ return กลับมานั้น เราอาจจะทำให้ set ค่าในตัวแปร string ของเราดอนแรกเพื่อทดสอบว่าค่าที่ return</p>	<pre>#include &lt;stdio.h&gt; #include &lt;string.h&gt;  char *GetText();  int main() {     char str[20];     strcpy( str, GetText() );     printf("%s", str);     return 0; }  char *GetText() {     char *text = new char[20];     printf("Enter your text : ");     gets(text);     return text; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

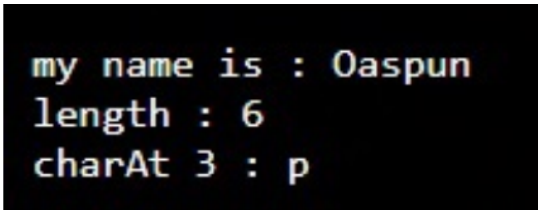
ข้อที่ 4 จงอธิบายความเกี่ยวข้องกันของ String และ Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>เราต้องสร้างตัวแปร string ที่ใช้ pointer 1</p> <p>ระดับก่อน พอตตอนเรียกใช้ void function ใน main ให้ได้ &amp; ไว้หน้าตัวแปรใน Argument</p> <p>ด้วยเพื่อเข้าถึง address ส่วนตัวแปรใน Argument ของ void function ที่รับค่าต้องใช้ pointer</p> <p>ระดับสูงกว่า ก็คือ</p> <p>จากนั้นถ้าต้องการจะทำอะไรกับตัวแปรใน void function ต้องลด pointer ลด 1 ระดับก็คือ *</p>	<pre>#include &lt;stdio.h&gt;  void GetText( char **text );  int main(){     char *str;     GetText(&amp;str);     printf("%s", str);     return 0; }  void GetText( char **text ) {     *text = new char[20];     printf("Enter your text : ");     gets(*text); }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre>graph TD     subgraph Process1 [ ]         direction TB         G1([GetText]) --&gt; S1[*str = new char[20]]         S1 --&gt; I1[/input *str/]         I1 --&gt; E1([**str])     end     subgraph Process2 [ ]         direction TB         S2([Start]) --&gt; S3[*str]         S3 --&gt; G2[GetText]         G2 --&gt; O2[/output str/]         O2 --&gt; E2([End])     end</pre>

ข้อที่ 5 จงอธิบายการการ Return ตัวแปรของ String แบบหลายๆ ข้อความ ข้อความละ 15 ตัวอักษร ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>สร้างตัวแปร char string 2 มิติใน main</p> <p>จากนั้นเราต้องสร้าง function ย่อยเป็น char pointer 2 มิติ แล้วใน function</p> <p>ย่อยเราจะประกาศตัวแปรเหมือนใน main ต่อมาใช้ for loop ในการรับค่าข้อความจาก input เข้า char string 2 มิติ แล้ว return เป็นตัวแปรกลับไปหา main</p> <p>ดังนั้นตัวแปรใน main จะต้อง รับค่าจาก function</p> <p>ย่อยเช่น str = function (); ใน main เราอาจจะใช้ for loop เพื่อแสดงค่าที่รับจาก function ย่อย</p>	<pre>#include &lt;stdio.h&gt;  char (*GetText2D( int row ))[16];  int main() {     char (*str)[16];     int row;     printf("How many rows do you want ? : ");     scanf("%d", &amp;row);     str = GetText2D(row);     for ( int i = 0 ; i &lt; row ; i++ ){         printf("%s \n", str[i]);     }      return 0; }  char (*GetText2D( int row ))[16]{     int i;     char (*text)[16];     for ( i = 0 ; i &lt; row ; i++ ){         printf("Enter your string [%d] : ", i);         scanf("%s", text[i]);     }      return text; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<pre>How many rows do you want ? : 8 Enter your string [0] : one Enter your string [1] : two Enter your string [2] : tree Enter your string [3] : four Enter your string [4] : five Enter your string [5] : six Enter your string [6] : seven Enter your string [7] : egg one two tree four five six seven egg</pre>	<pre>graph TD     subgraph GetText2D_Flowchart         Start1([GetText (int row)]) --&gt; Init1[(*str)[16] row]         Init1 --&gt; i0[i = 0]         i0 --&gt; Loop1{ i &lt; row ? }         Loop1 -- True --&gt; Input1[/input text/]         Input1 --&gt; Inc1[i++]         Inc1 --&gt; Loop1         Loop1 -- False --&gt; Return1([return text])     end      subgraph Main_Flowchart         Start2([Start]) --&gt; Init2[(*str)[16] row]         Init2 --&gt; Input2[/input row/]         Input2 --&gt; GetText2D[GetText2D(row)]         GetText2D --&gt; str[str = GetText2D(row)]         str --&gt; Loop2{ i &lt; row ? }         Loop2 -- True --&gt; Output2[/output str/]         Output2 --&gt; Inc2[i++]         Inc2 --&gt; Loop2         Loop2 -- False --&gt; End2([End])     end</pre>

## ตอนที่ 2 ทำความรู้จักกับสตริงในหลายๆ ภาษา

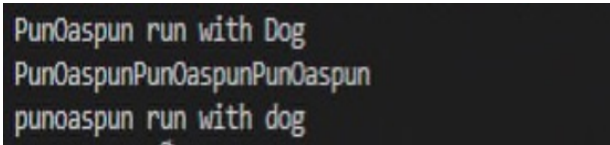
จงเขียนอธิบายการประกาศตัวแปรแบบ String ในภาษาต่างๆ ต่อไปนี้ อธิบายลักษณะการเก็บข้อมูล อธิบายฟังก์ชันที่เกี่ยวข้องกับ String ในภาษาเหล่านั้น พร้อมยกตัวอย่างประกอบการใช้งานให้ชัดเจน

ข้อที่ 1 String ในภาษา C#	
อธิบายการประกาศตัวแปร, ลักษณะการเก็บข้อมูล, ฟังก์ชันที่เกี่ยวข้อง	Code ตัวอย่าง
<p>Character เป็นประเภทข้อมูลที่ใช้ในการเก็บข้อมูลตัวอักษร โดยการใช้คำสั่ง char หรือ string ในการประกาศตัวแปร char จะเก็บตัวอักษรแบบ Unicode character ขนาด 16 bit string ใช้เก็บตัวอักษรหลายตัวหรือ text ฟังก์ชันที่เกี่ยวข้องได้แก่</p> <ul style="list-style-type: none"> <li>-trim() ใช้ลบ white spaces ที่จุดเริ่มและจุดจบ</li> <li>-ToUpper() ทำให้ตัวอักษรเป็นพิมพ์ใหญ่ทั้งหมด</li> </ul>	<pre>using System;  namespace HelloWorld {     class Program     {         static void Main(string[] args)         {             char shotname = '0';             string fullname = "Opaspun Klinchuenjit";             Console.WriteLine("My shotname is {0}", shotname);             Console.WriteLine("My fullname is {0}", fullname);             Console.WriteLine("My fullname is {0} ", fullname.ToUpper());         }     } }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
	

ข้อที่ 2 String ในภาษา Java	
อธิบายการประกาศตัวแปร, ลักษณะการเก็บข้อมูล, ฟังก์ชันที่เกี่ยวข้อง	Code ตัวอย่าง
<p>จะต้องประกาศ string ก่อนแล้วตามด้วยชื่อตัวแปรของเรา</p> <p>ชื่อตัวแปรห้ามขึ้นต้นด้วยตัวเลขและคำสงวน เราสามารถเก็บข้อมูลโดยใช้ =</p> <p>หลังชื่อตัวแปรตอนประกาศได้เลย เช่น string test = dog is big</p>	<pre> public class Main {     public static void main (String[] args) {         String a = "Oaspun";         System.out.println ("My name is : " + a);         System.out.println ("length : " + a.length);         System.out.println ("charAt 3: " + a.charAt(3));     } } </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
	

ข้อที่ 3 String ในภาษา PHP	
อธิบายการประกาศตัวแปร, ลักษณะการเก็บข้อมูล, ฟังก์ชันที่เกี่ยวข้อง	Code ตัวอย่าง
<p>ใน ภาษา php</p> <p>เราไม่ต้องประกาศประเภทตัวแปรแค่ใส่ \$</p> <p>หน้าชื่อตัวแปรของเรา</p> <p>มันจะรู้ชนิดตามประเภทข้อมูลที่อยู่ในตัวแปร</p> <p>เช่น \$big = "dog";</p> <p>ซึ่งการเก็บข้อมูลใส่ข้อมูลที่ต้องการไว้หลัง</p> <p>= ของตัวแปรแต่ถ้าเป็น String ต้องมี "" ด้วย</p>	<pre> &lt;!DOCTYPE html&gt;  &lt;html&gt; &lt;body&gt;  &lt;?php     \$name = "Opaspun";     echo "\$name go to school";     echo "&lt;br/&gt;";     echo var_dump(\$name); ?&gt;  &lt;/body&gt; &lt;/html&gt; </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
<p>Opaspun go to school</p> <p>string(7) "Opaspun"</p>	



ข้อที่ 4 String ในภาษา Java Script	
อธิบายการประกาศตัวแปร, ลักษณะการเก็บข้อมูล, ฟังก์ชันที่เกี่ยวข้อง	Code ตัวอย่าง
<p>การประกาศตัวแปรสามารถใช้ let</p> <p>แล้วตามด้วยชื่อตัวแปรของเราได้เลย</p> <p>ตัวแปรมันจะรู้เองว่าเป็นข้อมูลชนิดไหน เช่น let dog,</p> <p>let a ,let num การเก็บข้อมูลของเราไว้หลัง = ถ้าเป็น</p> <p>string ต้องมี "" ด้วยเช่น let dog = "red";</p>	<pre> let a = "PunOaspun"; let b = " run with Dog"; console.log(a + b); console.log(a.repeat(3)); console.log(`\${a} and friends \${b}`.toLowerCase()); </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
	

ข้อที่ 5 String ในภาษา Python	
อธิบายการประกาศตัวแปร, ลักษณะการเก็บข้อมูล, ฟังก์ชันที่เกี่ยวข้อง	Code ตัวอย่าง
<p>ใน ภาษา python</p> <p>แค่ประกาศชื่อตัวแปรก็สามารถใช้งานได้โดยไม่ต้องทำอะไรเพิ่ม การเก็บข้อมูลก็สามารถเก็บตรงๆได้เลย แต่ถ้าเป็นข้อความต้องมี "" หลัง = เอาคุม string เช่น</p> <p>fullname = "vave" ไม่ต้องมี; ด้วยถ้าเก็บแบบ Array หรือ List ก็จะเป็น zoo = ["dog", "cat"]</p>	<pre> fullname = "Opaspun Klinchuenjit" print(fullname)  animals = ['dog', 'cat', 'fish', 'bird', 'snake'] for i in range(0, len(animals)):     print(animals[i].upper(), end = ' ') </pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	
