

แผนการสอนประจำบทที่ 8

ตัวอักษรและสายอักขระ

หัวข้อสำคัญ

1. สายอักขระ (String) และอาร์เรย์ (Array)
2. ฟังก์ชันมาตรฐานของสายอักขระ
3. พอยน์เตอร์ของสายอักขระ

วัตถุประสงค์เชิงพฤติกรรม

1. ผู้เรียนสามารถอธิบายความหมายของสายอักขระและอาร์เรย์ได้
2. ผู้เรียนสามารถบอกชนิดของอาร์เรย์ได้
3. ผู้เรียนสามารถเขียนโปรแกรมประกาศตัวแปรอาร์เรย์และการอ้างอิงค่าจากอาร์เรย์ได้
4. ผู้เรียนสามารถเขียนโปรแกรมอาร์เรย์โดยการประยุกต์ใช้กับฟังก์ชันได้
5. ผู้เรียนสามารถเขียนโปรแกรมอาร์เรย์โดยใช้พอยน์เตอร์ได้

วิธีการสอนและกิจกรรมการเรียนการสอน

1. การบรรยาย
2. การทำแบบฝึกหัด

สื่อที่ใช้ประกอบการสอน

1. เอกสารประกอบการสอน
2. เครื่องคอมพิวเตอร์
3. เครื่องฉายภาพนิ่ง

การวัดและประเมินผล

1. สังเกตจากความสนใจของผู้เรียน
2. ประเมินจากการตอบคำถามของผู้เรียนและกิจกรรมในชั้นเรียน
3. การทำแบบฝึกหัดท้ายบท

บทที่ 8

ตัวอักษรและสายอักขระ

8.1 สายอักขระ (String)

ตัวแปรสายอักขระ (String) ในภาษา C จะใช้ Array ของ Character มาสร้างเป็นข้อความ ซึ่งความแตกต่างระหว่าง String และ Array ของ Character คือจะมี NULL Character(\0) ปรากฏอยู่ตำแหน่งสุดท้ายของข้อความ การจำลองการเก็บข้อมูลของตัวแปรสายอักขระชื่อ data แสดงดังภาพ 8.1

```
char data[15]="Hello world";
```

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]
H	e	l	l	o		W	o	r	l	d	\0	?	?	?

ภาพ 8.1 จำลองการเก็บข้อมูลของตัวแปรสายอักขระ

ข้อมูลที่อยู่ตำแหน่งสุดท้าย คือ NULL character(\0) ความยาวของข้อความคือ 11 ตัวอักษร

8.1.1 ตัวอย่างการประกาศตัวแปรสายอักขระ c

ตัวอย่างการประกาศตัวแปร

```
char c[] = "abcd";
```

```
char c[50] = "abcd";
```

```
char c[] = {'a', 'b', 'c', 'd', '\0'};
```

ผลลัพธ์จากการประกาศตัวแปร

c[0]	c[1]	c[2]	c[3]	c[4]
a	b	c	d	\0

สายอักขระ (String) และอาร์เรย์ของอักขระ (Array of Character)

สายอักขระ (String)

```
char color[] = "green";
```

```
char color[] = {'g', 'r', 'e', 'e', 'n', '\0'};
```

อาร์เรย์ของอักขระ

(Array of Character)

```
char color[] = {'g', 'r', 'e', 'e', 'n'};
```

8.1.2 กำหนดค่าเริ่มต้นให้กับตัวแปร String

การกำหนดแบบระบุความยาวตัวอักษรใน String

```
char str[10] = "Computer";
```

```
char str [10] = {'C','o','m','p','u','t','e','r','\0'};
```

การกำหนดแบบไม่ได้กำหนดความยาวของตัวอักษรใน String

```
char str[ ] = "Computer";
```

```
char *str = "Computer";
```

```
char str[] = {'C','o','m','p','u','t','e','r','\0'};
```

8.1.3 การคัดลอกหรือส่งค่าของตัวแปรชนิด String

ตัวแปรชนิด String ไม่สามารถใช้เครื่องหมาย = เหมือนคำสั่งสำหรับข้อมูลชนิดอื่นได้ เพื่อคัดลอกค่าไปยังตัวแปรอื่นได้ ตัวอย่างแสดงดังภาพ 8.2

Integer	String
<pre>int number1, number2 ; number1 =10; number2 = number1;</pre> <p>ผลลัพธ์</p>	<pre>char str1[10]="abcABC",str2[10]; str1[10] = str2[10]; printf("string1 = %s\n",str1); printf("string2 = %s\n",str2); strcpy(str2,str1); printf("string1 = %s\n",str1); printf("string2 = %s\n",str2);</pre> <p>ผลลัพธ์</p>

ภาพ 8.2 ตัวอย่างการคัดลอกค่าของตัวแปรจำนวนเต็มและสายอักขระ

8.2. ฟังก์ชันมาตรฐานของสายอักขระ

ตัวอย่างฟังก์ชันที่ประกาศในเฮดเดอร์ไฟล์ string.h

ชื่อฟังก์ชัน	การทำงาน
void strcpy(str1, str2);	คัดลอก (copy) ข้อความจาก str2 มาไว้ที่ str1
void strcat(str1, str2);	นำเอาข้อความ str2 มาต่อท้าย str1 แล้วเก็บค่าที่ได้ไว้ใน str1
int strcmp(str1, str2);	เปรียบเทียบข้อความ ผลลัพธ์เป็น 0 ถ้า str1 == str2, <0 ถ้า str1 < str2 และ >0 ถ้า str1 > str2
int strcmpi(str1, str2);	เหมือนกับ strcmp แต่ไม่สนใจเรื่อง case
int strlen(str);	หาความยาวของข้อความ

ตัวอย่างที่ 1 การใช้ฟังก์ชัน string.h

โปรแกรม	ผลลัพธ์
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() { 4. char str1[] = "computer"; 5. char str2[] = "Computer"; 6. char str3[20]; 7. strcpy(str3, str1); 8. strcat(str3, str2); 9. printf("%s\n\n", str3); 10. printf("%d\n\n", strcmp(str1, str2)); 11. printf("%d\n\n", strcmpi(str1, str2)); 12. printf("%d", strlen(str1)); 13. }</pre>	

ตัวอย่างที่ 2 การใช้ฟังก์ชัน strcpy

โปรแกรม	ผลลัพธ์
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. char str1[10]= "Hello"; 6. char str2[10], str3[10]; 7. strcpy(str2, str1); 8. strcpy(str3, "C language"); 9. puts(str2); 10. puts(str3); 11. }</pre>	

ตัวอย่างที่ 3 การใช้ฟังก์ชัน strcat

โปรแกรม	ผลลัพธ์
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. char str1[10]= "Hello"; 6. char str2[10] ="C language"; 7. strcat(str1,str2); 8. puts(str1); 9. strcat(str1,"\tworld"); 10. puts(str1); 11. }</pre>	

ตัวอย่างที่ 4 การใช้ฟังก์ชัน strcmp

โปรแกรม	ผลลัพธ์
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. char str1[] = "abcd", str2[] = "abCd"; 6. char str3[] = "abcd"; 7. int result_cmp; 8. result_cmp = strcmp(str1, str2); 9. printf("strcmp(str1, str2) = %d\n", result_cmp); 10. result_cmp = strcmp(str1, str3); 11. printf("strcmp(str1, str3) = %d\n", result_cmp); 12. result_cmp = strcmp(str2, str3); 13. printf("strcmp(str2, str3) = %d\n", result_cmp); 14. } 15. </pre>	

ตัวอย่างที่ 5 การใช้ฟังก์ชัน strcmpi

โปรแกรม	ผลลัพธ์
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. char str1[] = "abcd", str2[] = "abCd"; 6. char str3[] = "abcd"; 7. int result_cmp; 8. result_cmp = strcmpi(str1, str2); 9. printf("strcmp(str1, str2) = %d\n", result_cmp); 10. result_cmp = strcmpi(str1, str3); 11. printf("strcmp(str1, str3) = %d\n", result_cmp); 12. } </pre>	

ตัวอย่างที่ 6 การใช้ฟังก์ชัน strlen

โปรแกรม
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. char str1[20]="Program"; 6. char str2[20]='P','r','o','g','r','a','m','\0'; 7. char str3[20]; 8. printf("Length of string str1 = %d \n",strlen(str1)); 9. printf("Length of string str2 = %d \n",strlen(str2)); 10. printf("Length of string str3 = %d \n",strlen(str3)); 11. }</pre>
ผลลัพธ์

8.3. พอยน์เตอร์ของสายอักขระ (Pointer of String)

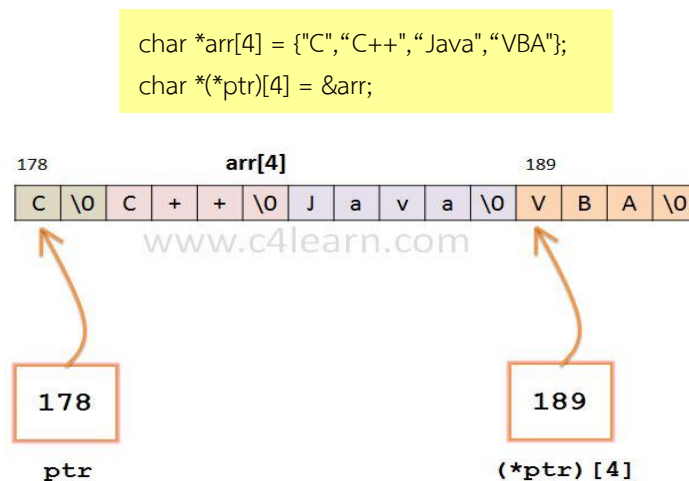
8.3.1. การแสดงค่าและตำแหน่งของพอยน์เตอร์ของสายอักขระ (Pointer of String)

ตัวแปรชนิดสายอักขระ (String) %u แสดงตำแหน่งที่อยู่ในหน่วยความจำเป็นเลขจำนวนเต็มไม่มีเครื่องหมายและเป็นเลขฐานสิบ

ตัวอย่างที่การแสดงค่าและตำแหน่งข้อมูลที่พอยน์เตอร์ชี้

โปรแกรม
<pre> 1. #include <stdio.h> 2. main() 3. { 4. int i; 5. char data[10] = "ABCDEF"; 6. char *ptr; 7. for(i=0;i<4;i++) 8. { 9. ptr = &data[i]; 10. printf("%d Address of String %c : %u\n",i+1,*ptr, ptr); 11. } 12. }</pre>
ผลลัพธ์

พอยน์เตอร์ที่ชี้ไปยังข้อมูลที่เก็บเป็นอาร์เรย์ของสายอักขระ เรียกว่า **Pointer to array of string** ที่มา <http://www.c4learn.com/c-programming/c-pointer-to-array-of-string/> การจำลองการเก็บข้อมูลอาร์เรย์ของสายอักขระดังภาพ 8.3



ภาพ 8.3 : จำลองการเก็บข้อมูลอาร์เรย์ของสายอักขระ

ตัวอย่างที่ 1 การแสดงค่าและตำแหน่งของอาร์เรย์ของสายอักขระ

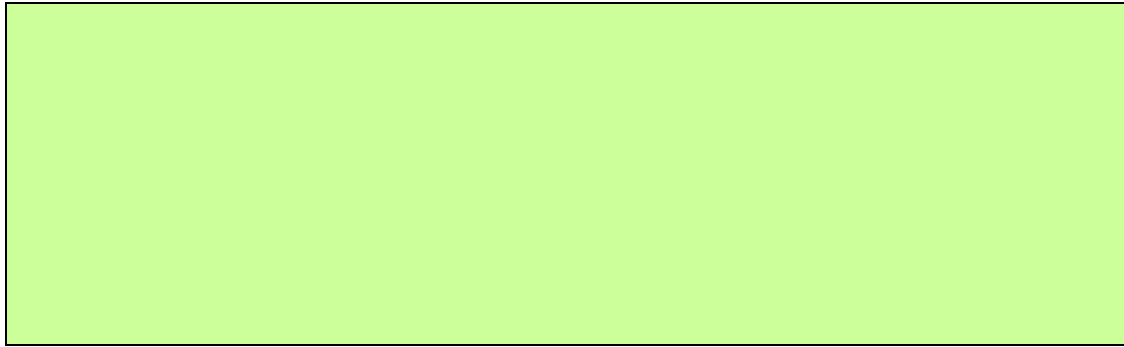
โปรแกรม	ผลลัพธ์
<pre> 1. #include<stdio.h> 2. main() 3. { 4. int i; 5. char *arr[4] = {"C","C++","Java","VBA"}; 6. char *(*ptr)[4] = &arr; 7. for(i=0;i<4;i++) 8. printf("Address of String %d : %u\n",i+1,(*ptr)[i]); 9. }</pre>	<p>อ้างอิงที่อยู่สมมุติของตัวแปร *arr[4]</p>

ตัวอย่างที่ 2 การแสดงค่าและตำแหน่งของอาร์เรย์ของสายอักขระ

โปรแกรม	ผลลัพธ์
<pre> 1. #include<stdio.h> 2. main() 3. { 4. int i; 5. char *arr[4] = {"C","C++","Java","VBA"}; 6. char *(*ptr)[4] = &arr; 7. for(i=0;i<4;i++) 8. printf("Address of String %d : %s\n",i+1,(*ptr)[i]); 9. }</pre>	<p>อ้างอิงที่อยู่สมมุติของตัวแปร *arr[4]</p>

ตัวอย่างที่ 3 การแสดงค่าและตำแหน่งของอาร์เรย์ของสายอักขระ

โปรแกรม	ผลลัพธ์
<pre> 1. #include<stdio.h> 2. main() 3. { 4. int i; 5. char *arr[4] = {"C","C++","Java","VBA"}; 6. char *(*ptr)[4] = &arr; 7. for(i=0;i<4;i++) 8. printf("%d Address of String %s : %u\n",i+1,(*ptr)[i],(*ptr)[i]); 9. }</pre>	
	ผลลัพธ์



8.3.2. การเลื่อนตำแหน่งการชี้ของพอยน์เตอร์

ตัวอย่างการเลื่อนตำแหน่งการชี้ของพอยน์เตอร์

โปรแกรม	ผลลัพธ์
<pre> 1. #include<stdio.h> 2. main() 3. { 4. int i; 5. char *arr[4] = {"C","C++","Java","VBA"}; 6. char *(*ptr)[4] = &arr; 7. printf("%s\n",(*ptr)[2]); 8. printf("%s\n",(*ptr)[2]); 9. printf("%s\n",++(*ptr)[2]); 10. printf("%s\n",(*ptr)[2]++); 11. printf("%s\n",(*ptr)[2]); 12. }</pre>	

8.3.3 การสร้างฟังก์ชันของพอยน์เตอร์

ตัวอย่างการสร้างฟังก์ชันของพอยน์เตอร์

โปรแกรม	ผลลัพธ์
<pre>1. #include<stdio.h> 2. void display(); 3. main() 4. { 5. void *(*ptr)(); 6. ptr = &display; 7. (*ptr)(); 8. } 9. void display() 10. { 11. printf("C Programming"); 12. }</pre>	

คำถามท้ายบทที่ 8

1. จงอธิบายความแตกต่างของการประกาศตัวแปรระหว่างที่กำหนดความยาวตัวอักษรและไม่ได้กำหนดความยาวตัวอักษร พร้อมทั้งยกตัวอย่างประกอบ
2. จงอธิบายความแตกต่างระหว่างตัวแปรชนิดสายอักขระ (String) และอาร์เรย์ของอักขระ (Array of Character)
3. จงอธิบายความหมายและยกตัวอย่างการทำงานของคำสั่ง strcpy
4. จงอธิบายความหมายและยกตัวอย่างการทำงานของคำสั่ง strcat
5. จงอธิบายความหมายและยกตัวอย่างการทำงานของคำสั่ง strcmp และ strcmpi
6. จงอธิบายความแตกต่างของการประกาศตัวแปรระหว่างที่กำหนดความยาวตัวอักษรและไม่ได้กำหนดความยาวตัวอักษร พร้อมทั้งยกตัวอย่างประกอบ
7. จงเขียนโปรแกรมให้รับข้อมูล String จำนวน 2 ค่าจากแป้นคีย์บอร์ด และใช้ฟังก์ชันจาก string.h โดยกำหนดให้แสดงผลดังตัวอย่างด้านล่าง

```
Enter a string 1: ABC
Enter a string 2: computer

New string = ABCcomputer
New string = computerABCcomputer
New string = ABCcomputer
New string = ABCcomputerABCcomputer
Length of string3: 22_
```

8. จงเขียนโปรแกรมค้นหาและแสดงจำนวนของอักขระที่เกิดจากการรับข้อมูล String จำนวน 2 ค่าจากแป้นคีย์บอร์ด และใช้ฟังก์ชันจาก string.h เพื่อเชื่อมข้อมูลที่ได้รับเข้าพร้อม และทั้งรับค่าตัวอักษรที่ใช้ค้นหา โดยกำหนดให้แสดงผลดังตัวอย่างด้านล่าง

```
Enter a string 1: Computer
Enter a string 2: Information
All string = ComputerInformation

*****
Find a character : m
Number of m = 2_
```

9. จงหาผลลัพธ์จากโปรแกรมดังต่อไปนี้

โปรแกรม	ผลลัพธ์(แสดงผลลัพธ์ตามบรรทัดบนหน้าจอคอมพิวเตอร์)
<pre> 1. #include<stdio.h> 2. #include<string.h> 3. main() 4. { 5. char ch[10]={'h', 'e', 'l', 'l', 'o', '\0'}; 6. char ch2[10]={'c', '\0'}; 7. strcat(ch,ch2); 8. printf("%s",ch); 9. 10. char string1[4] = "ABC"; 11. char string2[4] = "abc"; 12. printf("%d",strcmp(string1, string2)); 13. 14. printf("%d\n", strlen("computer programming c++")); 15. }</pre>	<pre> 1. 2. 3.</pre>

10. จงหาผลลัพธ์จากโปรแกรมดังต่อไปนี้

โปรแกรม	ผลลัพธ์(แสดงผลลัพธ์ตามบรรทัดบนหน้าจอคอมพิวเตอร์)
<pre> 1. #include <stdio.h> 2. #include <string.h> 3. main() 4. { 5. int i; 6. char data1[25] = "Kasetsart university"; 7. char data2[25] = "Sriracha campus"; 8. char *ptr; 9. for(i=0;i<strlen(data2);i+=3) 10. { 11. ptr = &data1[i]; 12. printf("%d Address of String %c : %u\n",i+1,*ptr, ptr); 13. } 14. }</pre>	<pre> 1. 2. 3. 4. 5. 6. 7. 8.</pre>