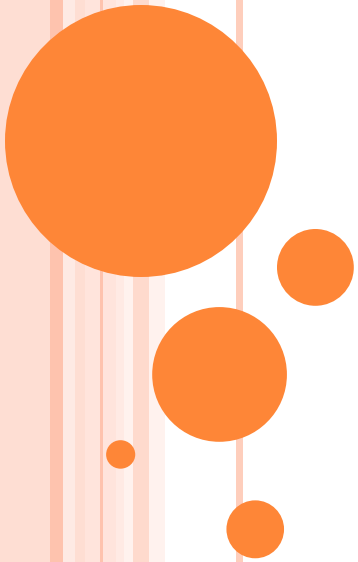

UNIT 6: STRINGS

Nguyen Thi Thu Huong
Department of Computer Science
Hanoi University of Technology

CHARACTER STRINGS

- A variable of type *char* may hold a single character
  - Example: char **onechar = 'z'** ;

- Character strings are *arrays* of simple characters with a special character inserted into the string **at the very end**: *null* character ('\0').

- They are assigned values with a pair of double quotes:
  - Example: char **arraychar[6] = "abcde"** ;

Nguyen Thi Thu Huong

Character Strings

- Strings are normally accessed by a pointer to the first character in the string.
- This means that the value of a string is the address of it's first character.

Nguyen Thi Thu Huong

CHARACTER STRINGS

- Declaration and initialization:

  char **color [ ] = "scarlet"** ;

  or

  char ***colorPtr = "scarlet"** ;

  or

  char **color [8] = {'s', 'c', 'a', 'r', 'l', 'e', 't', '\0'}** ;

- NOTE: Allowance **must always** be made for the terminating null character.

Nguyen Thi Thu Huong

String I/O Library Routines

- #include <stdio.h>

/* The following are *function prototypes* for some of the String I/O and Handling Library Routines */

/* Input next character as an integer */

int getchar (void) ;

(Don't confuse with getch() of conio.h)

/* Input string into array s until newline */

char *gets (char *s) ;

Nguyen Thi Thu Huong

String I/O Library Routines

- #include <stdio.h>

/* Print character stored in character variable c */

int putchar (int c) ;

(Don't confuse with putch() of conio.h)

/* Print character string s followed by \n */

int puts (const char *s) ;

Character classification functions

Category	Functions in ctype.h
Letters	isalpha()
Lowercase letters	islower()
Uppercase letters	isupper()
Decimal digits	isdigit()
Hexadecimal digits	isxdigit()
Letters and decimal digits	isalnum()

STRING-PROCESSING FUNCTIONS

Purpose	Functions in string.h
Find the length of a string.	strlen()
Copy a string.	strcpy()
Concatenate strings.	strcat()
Compare strings.	strcmp( )
In a string, find:	
• The first or last occurrence of a given character	strchr()
• The first occurrence of another string	strstr()

STRING-PROCESSING FUNCTIONS

- **size_t strlen(char[] s)**
  - Get length of string s
  - printf("%d ",strlen("Hello world")); ⇒ 11
- **char[] strcpy(char[] str1, char[] str2)**
  - Copy string string str2 to string str1
  - printf("%s ",strcpy(Str,"Hello")); ⇒ Hello
  - printf("%s", Str); ⇒ Hello
- **int strcmp(char[] str1, char[] str2)**
  - Compare str1 to str2
  - Return 0 if they are equal to each other;
  - Return a value < 0: str1 < str2
  - Return a value >0: str1 > str2

String-processing functions

- **char[] strcat(char[] str2, char[] str2)**
  - Appends a copy of the string str2 to the end of the string str1. ...
  - The strcat function returns a pointer to str1
  - Example

```
char Str[20];
strcpy(Str,"Hello ");
printf("%s ",strcat(Str,"world")); ⇒ Helloworld
printf("\n%s",Str); ⇒ Helloworld
```

Nguyen Thi Thu Huong

String-processing functions

- **char * strchr (char * s, int c)**
  - Searches for the first occurrence of the character c (an unsigned char) in the string pointed to by the argument s
  - . strcpy(Str,"Hello world");

    printf("%s ",strchr(Str,'o')); ⇒ o world
- **char* strstr(char * s1, char * s2)**
  - Return a pointer to the first occurrence in str1 of the entire sequence of characters specified in str2, or a null pointer if the sequence is not present in str1.

    printf("%s ",strstr(Str,"llo")); ⇒ llo world

Nguyen Thi Thu Huong

SEARCHING USING STRSTR

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
main()
{char s[20],u[20];
int m,i, n;
puts("Enter your string");gets(s);
fflush(stdin);
puts("Enter string you want to search");gets(u);
if(strstr(s,u)==NULL)puts("False");else
puts("True");
}
```

Nguyen Thi Thu Huong

SEARCHING

Enter two strings *s* and *u* and an integer number *n* The program prints the position of the first occurrence of u within a substring of s. n is the position where the substring begins. If *t* is not found in the substring of *s*, the program prints zero

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
main()
{char s[20],t[20],u[20];
int m,i, n;
puts("Enter your string");gets(s);
puts("n=");scanf("%d",&n);
m=strlen(s);
for(i=n;i<=m;i++) t[i-n]=s[i];
fflush(stdin);
puts("Enter string you want to search");gets(u);
if(strstr(t,u)==NULL)puts("0");else
{printf("\nPosition of the first occurence of string %s in a substring of %s",u,s);
printf("\nstart from %dth character is %d",n, strstr(t,u)-&t[0]+n);}
}
```

Nguyen Thi Thu Huong

CLASSICAL EXAMPLE: CONVERT A NUMBER FROM DECIMAL TO BINARY

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
main()
{char s[100]="",t[100];
int i, n;
puts("n="); scanf("%d",&n);
i=n;
while (i>0)
{if(i%2==0) strcpy(t,"0");else strcpy(t,"1");
strcat(t,s);
strcpy(s,t);
i/=2;}
printf("\n");
puts(s);
}
```

Nguyen Thi Thu Huong