



4. Xâu ký tự – string

Nội dung

- Xâu ký tự
- Lưu trữ xâu ký tự
- Khởi tạo và in nội dung xâu
- Nhập xâu từ bàn phím
- Một số ví dụ
- Một số hàm trong thư viện ctype.h và string.h

4 Xâu ký tự – string

■ `printf ("Programming in C is fun.\n");`

`"Programming in C is fun.\n"`

■ **Xâu ký tự:**

- là một dãy các ký tự,
- Các ký tự có thể là chữ cái, chữ số hoặc cả các ký tự đặc biệt
- Xâu ký tự được đặt trong dấu nháy kép " "
- Xâu ký tự không có ký tự nào là xâu rỗng (viết là "")

4 Xâu ký tự – string

■ **Mảng ký tự**

`char word[] = { 'H', 'e', 'l', 'l', 'o', '!' };`

■ Để in ra xâu ta in lần lượt từng ký tự

```
for ( i = 0; i < 6; ++i )
    printf ("%c", word[i]);
```

- ➔ **Nhược điểm:** Phải luôn lưu thông tin về số lượng lý tự trong mảng ký tự

| | |
|---------|-----|
| word[0] | 'H' |
| word[1] | 'e' |
| word[2] | 'l' |
| word[3] | 'l' |
| word[4] | 'o' |
| word[5] | '!' |

Mảng word trong bộ nhớ

4 Xâu ký tự – string

- **Giải pháp:** sử dụng một ký tự đặc biệt (ký tự null) báo hiệu kết thúc xâu: `'\0'`

```
const char word [] = { 'H', 'e', 'l', 'l', 'o', '!', '\0' };
```

- Tìm độ dài của xâu

```
int count = 0;
while ( str[count] != '\0' )
    ++count;
```

| | |
|---------|------|
| word[0] | 'H' |
| word[1] | 'e' |
| word[2] | 'l' |
| word[3] | 'l' |
| word[4] | 'o' |
| word[5] | '!' |
| word[6] | '\0' |

4 Xâu ký tự – string

- Khởi tạo

```
char tên_biến[độ_dài]={nội_dung};
```

- Ví dụ

```
char word[] = { "Hello!" };
```

```
char word[] = "Hello!";
```

```
char word[] = { 'H', 'e', 'l', 'l', 'o', '!', '\0' };
```

```
char word[10] = { "Hello!" };
```

```
char word[6] = { "Hello World!" }; ???
```

4 Xâu ký tự – string

In nội dung:

- Hàm `printf`

- Hàm `printf` xác định kết thúc xâu bằng ký tự null

```
printf("Programming in C is fun.\n");
```

- In nội dung mảng ký tự kết thúc bằng ký tự null (`'\0'`)

```
printf("%s\n", word);
```

- Hàm `puts`

```
puts(tên_biến);
```

4 Xâu ký tự – string

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    int i = 0;
```

```
    char str[30] = "This is a sample";
```

```
    char word[] = { 'H', 'e', 'l', 'l', 'o', '!', '\0' };
```

```
    printf("Nội dung các xâu\n");
```

```
    printf("xau str: %s\n", str);
```

```
    printf("xau word: %s\n", word);
```

```
    for(i=0; i<4; i++) printf("%c", str[i]);
```

```
    return 0;
```

```
}
```

4 Xâu ký tự – string

Nhập xâu từ bàn phím

■ Hàm scanf()

```
char string[81];
scanf("%s", string);
```

```
int main (void)
{
    char s1[81], s2[81], s3[81];
    printf ("Enter text:\n");
    scanf ("%s%s%s", s1, s2, s3);
    printf ("\ns1 = %s\ns2 = %s\ns3 = %s\n",
        s1, s2, s3);
    return 0;
}
```

4 Xâu ký tự – string

■ Hàm gets

```
gets(tên_biến);
```

```
int main (void)
{
    char s1[81], s2[81], s3[81];
    printf ("Nhap xau 1:"); gets(s1);
    printf ("Nhap xau 2:"); gets(s2);
    printf ("Nhap xau 3:"); gets(s3);

    printf ("\ns1 = %s\ns2 = %s\ns3 = %s\n",
        s1, s2, s3);
    return 0;
}
```

4 Xâu ký tự – string

- **VD1.** Viết chương trình nhập nội dung xâu từ bàn phím bằng cách nhập lần lượt từng ký tự trong xâu sử dụng hàm **getchar()**

```
char character, buffer[81];
int i = 0;
printf("Nhap xau:");
do
{
    character = getchar();
    buffer[i] = character;
    ++i;
}
while ( character != '\n' );
buffer[i - 1] = '\0';
printf("Noi dung xau: %s\n",buffer);
```

4 Xâu ký tự – string

- **VD2.** Ghép nội dung hai xâu s1 và s2 lại với nhau để tạo thành xâu mới.

```
int i, j;

// copy noi dung s1 vao result
for ( i = 0; s1[i] != '\0'; ++i )
    result[i] = s1[i];

// copy noi dung s2 vao result
for ( j = 0; s2[j] != '\0'; ++j )
    result[i + j] = s2[j];

// them ky tu ket thuc vao xau ket qua
result [i + j] = '\0';
```

4 Xâu ký tự – string

- **VD3.** So sánh hai xâu s1 và s2 có bằng nhau hay không. Hai xâu bằng nhau là hai xâu có các ký tự tại các vị trí tương ứng giống nhau.

```
int areEqual;

while ( s1[i] == s2[i] &&
s1[i] != '\0' && s2[i] != '\0' )
    ++i;

if ( s1[i] == '\0' && s2[i] == '\0' )
    areEqual = 1;
else
    areEqual = 0;
```

4 Xâu ký tự – string

```
printf ("\aWARNING!!\n");

printf ("%i\t%i\t%i\n", a, b, c);

printf ("\t is the horizontal tab
character.\n");

printf ("\\"Hello,\" he said.\n");
```

| Ký hiệu | Ý nghĩa |
|---------|-----------------|
| \a | Audible alert |
| \b | Backspace |
| \f | Form feed |
| \n | Newline |
| \r | Carriage return |
| \t | Horizontal tab |
| \v | Vertical tab |
| \\ | Backslash |
| \" | Double quote |
| \' | Single quote |
| \? | Question mark |

Một số ký tự đặc biệt

4 Xâu ký tự – string

- Một số hàm trong thư viện ctype.h

| Hàm | Ý nghĩa | Ví dụ |
|-------------|---|-----------------------|
| isalnum(c) | Kiểm tra một ký tự có phải là chữ cái hoặc số hay không (A-Z, a-z, 0-9) | isalnum ('a') → TRUE |
| isalpha(c) | Kiểm tra một ký tự có phải là chữ cái hay không (A-Z, a-z) | isalpha ('3') → FALSE |
| isdigit(c) | Kiểm tra một ký tự có phải là chữ số hay không (0-9) | |
| islower(c) | Kiểm tra một chữ cái là chữ thường (a-z). | |
| isupper(c) | Kiểm tra một chữ cái là chữ hoa (A-Z). | |
| isspace(c) | Kiểm tra ký tự có phải là dấu cách trống hay không | |
| isxdigit(c) | Kiểm tra ký tự có phải là chữ số trong hệ hexa hay không (0-9, A-F, a-f). | |
| toupper(c) | Chuyển ký tự chữ cái từ thường thành hoa (nếu không phải chữ cái thì không làm gì cả) | |
| tolower(c) | Chuyển ký tự chữ cái từ hoa thành thường (nếu không phải chữ cái thì không làm gì cả) | |

4 Xâu ký tự – string

- Một số hàm trong thư viện string.h

| Hàm | Ý nghĩa | Ví dụ |
|---------|--|---|
| strcat | Nối hai xâu ký tự lại làm một | char *strcat(char *dest, const char *src); |
| strncat | Nối hai xâu ký tự lại làm một (chỉ nối n ký tự đầu của xâu nguồn) | char *strncat(char *dest, const char *src, size_t n); |
| strlen | Trả về độ dài của một xâu ký tự (số ký tự có trong xâu) | char str[]="Hello!"; printf("%d",strlen(str)); |
| strcmp | So sánh hai xâu ký tự, trả về giá trị | if(strcmp(str,word)==0) printf("They are equal!"); |
| strcpy | Copy nội dung của xâu thứ hai vào xâu thứ nhất | strcpy(str,word); |
| strchr | Tìm kiếm một ký tự trong xâu. Nếu tìm thấy thì trả về vị trí xuất hiện đầu tiên | if(strchr(string, character)) puts("Character found"); |
| strstr | Tìm kiếm một xâu con trong xâu ký tự ban đầu. Hàm trả về con trỏ tới vị trí tìm được, nếu không tìm thấy thì trả về NULL | pointer = strstr(string1, string2); |

4. Xâu ký tự – string

- ▣ VD1. Viết chương trình nhập vào một xâu ký tự bất kỳ từ bàn phím. Đếm và in ra màn hình số lượng ký tự hoa, ký tự thường và chữ số trong xâu đó.
- ▣ VD2. Viết chương trình in ra màn hình nội dung một xâu ký tự nhưng theo thứ tự ngược.