

KỸ THUẬT LẬP TRÌNH C/C++

Chương 3: KỸ THUẬT LẬP TRÌNH TRÊN CHUỖI KÝ TỰ

Nội dung

1. Khái niệm chuỗi ký tự
2. Xuất nhập chuỗi ký tự
3. Kỹ thuật xử lý chuỗi

1. Khái niệm chuỗi ký tự

1. Khái niệm chuỗi ký tự

Chuỗi ký tự là một dãy gồm các ký tự hoặc một mảng các ký tự.

Chuỗi ký tự được đặt trong dấu nháy kép " ", ví dụ: "Hello World!"

Sử dụng các hàm để xử chuỗi ký tự như: tìm chiều dài chuỗi, so sánh hai chuỗi, tìm kiếm chuỗi, rút trích chuỗi con, nối chuỗi, ...

Khai báo một biến kiểu chuỗi theo cú pháp sau:

string <tên_biến> = **“Chuỗi ký tự”**;

Trong đó: string là kiểu chuỗi ký tự 32 bytes.

Lưu ý: Cần khai báo thư viện string.h ở đầu chương trình:

#include <string.h>

Ví dụ 1: Khai báo biến str là biến kiểu chuỗi và chứa chuỗi "Hello World!"

a) `string str = "Hello World!";`

b) `string str1 = "Hello World!";`

`string str2 = "I am learning C++";`

1. Khái niệm chuỗi ký tự

❖ Vị trí của các ký tự trong chuỗi

Một chuỗi với chiều dài n ký tự, vị trí của các ký tự trong chuỗi có phạm vi từ 0 đến $n - 1$.

Có thể truy cập ký tự trong chuỗi theo cú pháp như sau: **<biến kiểu chuỗi> [vị trí]**

Ví dụ 2: Khai báo biến `str` là biến kiểu chuỗi và chứa chuỗi "Hello World!", truy cập đến ký tự thứ 2 và ký tự thứ 7 trong chuỗi này.

```
string str = "Hello World!";
```

`str[0] str[1] str[2] ... str[6]`

Ký tự thứ 2 tương ứng vị trí trong chuỗi là: `str[1] = e`

Ký tự thứ 7 tương ứng vị trí trong chuỗi là: `str[6] = W`

2. Xuất nhập chuỗi ký tự

2. Xuất nhập chuỗi ký tự

❖ **Xuất chuỗi ký tự:** sử dụng lệnh `cout` có cú pháp như sau:

`cout << biến_kiểu_chuỗi;`

Ví dụ 2 (a):

```
#include <iostream>
#include <string.h>
using namespace std;
int main() {
    string str = "Hello World!";
    cout << str;
    return 0;
}
```

➤ Kết quả:

Hello World!

Ví dụ 2 (b):

```
#include <iostream>
#include <string.h>
using namespace std;
int main() {
    string str1 = "Hello World!";
    string str2 = "I am learning C++";
    cout << str1 << endl;
    cout << str2 << endl;
    return 0;
}
```

➤ Kết quả: Hello World!
I am learning C++

2. Xuất nhập chuỗi ký tự

❖ **Nhập chuỗi ký tự:** Chức năng nhập chuỗi ký tự từ bàn phím và lưu vào biến kiểu chuỗi, có 2 dạng cú pháp như sau:

`cin >> biến_kiểu_chuỗi;` chỉ lưu một từ duy nhất trước khoảng trắng (dù nhập vào nhiều từ), vì `cin` xem một khoảng trắng (khoảng trắng hoặc khoảng tab) là một ký tự kết thúc.

`getline (cin, biến_kiểu_chuỗi);` lưu một dòng văn bản (bao gồm nhiều từ và kể cả khoảng trắng hoặc tab). Hàm này sử dụng `cin` làm tham số đầu tiên và `biến_kiểu_chuỗi` là tham số thứ hai.

2. Xuất nhập chuỗi ký tự

❖ Nhập chuỗi ký tự (tt):

Ví dụ 3 (a):

```
#include <iostream>

using namespace std;

int main() {

    string name;

    cout << "Nhap vao ho va ten: ";

    cin >> name;

    cout << "Ho va ten da nhap la: " << name;

}
```

➤ Kết quả:

Nhap vao ho va ten: Nguyen Van An

Ho va ten da nhap la: **Nguyen**

Ví dụ 3 (b):

```
#include <iostream>

using namespace std;

int main() {

    string name;

    cout << "Nhap vao ho va ten: ";

    getline(cin,name);

    cout << "Ho va ten da nhap la: " << name;

}
```

➤ Kết quả:

Nhap vao ho va ten: Nguyen Van An

Ho va ten da nhap la: **Nguyen Van An**

3. Kỹ thuật xử lý chuỗi

3. Kỹ thuật xử lý chuỗi

1. Nối chuỗi

Trong ngôn ngữ lập trình C ++, toán tử “+” có thể được sử dụng giữa các chuỗi để nối chúng lại với nhau để tạo thành một chuỗi mới.

Ví dụ 4 (a):

```
#include <iostream>

using namespace std;

int main () {
    string str1 = "Gia";
    string str2 = "Dinh";
    string str3 = str1 + str2;
    cout << str3;
    return 0;
}
```

➤ Kết quả: GiaDinh

Ví dụ 4 (b):

```
#include <iostream>

using namespace std;

int main () {
    string str1 = "Gia ";
    string str2 = "Dinh";
    string str3 = str1 + str2;
    cout << str3;
    return 0;
}
```

➤ Kết quả: Gia Dinh

Ví dụ 4 (c):

```
#include <iostream>

using namespace std;

int main () {
    string str1 = "Gia";
    string str2 = "Dinh";
    string str3 = str1 + " " + str2;
    cout << str3;
    return 0;
}
```

➤ Kết quả: Gia Dinh

3. Kỹ thuật xử lý chuỗi

1. Nối chuỗi (tt)

Lưu ý: Cần phân biệt trong việc sử dụng toán tử “+” đối với số học và đối với chuỗi.

Ví dụ 5 (a): Cộng số học

```
#include <iostream>
using namespace std;
int main () {
    int a = 20;
    int b = 22;
    int c = a + b;
    cout << c;
    return 0;
}
```

➤ Kết quả: 42

Ví dụ 5 (b): Cộng chuỗi

```
#include <iostream>
using namespace std;
int main () {
    string a = "20";
    string b = "22";
    string c = a + b;
    cout << c;
    return 0;
}
```

➤ Kết quả: 2022

Ví dụ 5 (b): Cộng số với chuỗi sẽ xảy ra lỗi

```
#include <iostream>
using namespace std;
int main () {
    int a = 20;
    string b = "22";
    string c = a + b;
    cout << c;
    return 0;
}
```

➤ Kết quả: [Error] no match for 'operator+'

3. Kỹ thuật xử lý chuỗi

1. Nối chuỗi (tt)

Nối chuỗi sử dụng hàm `append ()`. Cú pháp: `chuỗi_1.append(chuỗi_2);`

Ví dụ 6:

```
#include <iostream>
using namespace std;
int main () {
    string str1 = "Gia ";
    string str2 = "Dinh";
    string str3 = str1.append(str2);
    cout << str3;
    return 0;
}
```

➤ Kết quả: Gia Dinh

3. Kỹ thuật xử lý chuỗi

2. Truy cập ký tự trong chuỗi

Truy cập các ký tự trong chuỗi str bằng cách tham chiếu đến số chỉ mục (vị trí) của nó bên trong dấu ngoặc vuông []. Vị trí ký tự đầu tiên trong chuỗi bằng [0], ký tự thứ hai bằng [1], ...

Cú pháp: **str[vị trí];** (với **str** là biến kiểu chuỗi)

Ví dụ 7: In ra màn hình ký tự ở vị trí 1 và vị trí 6 trong chuỗi “Hello World!”.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str = "Hello World!";
```

```
    cout << str[1] << endl;
```

```
    cout << str[6] << endl;
```

```
    return 0;
```

```
}
```

➤ Kết quả:

e

W

3. Kỹ thuật xử lý chuỗi

2. Truy cập ký tự trong chuỗi (tt)

Thay đổi ký tự trong chuỗi str: **str[vị trí] = 'Ký tự cần thay thế';** (với **str** là biến kiểu chuỗi)

Ví dụ 8: Thay đổi ký tự **W** trong chuỗi “Hello **W**orld!” bằng ký tự **H**.

```
#include <iostream>
using namespace std;
int main() {
    string str = "Hello World!";
    str[6] = 'H';
    cout << str;
    return 0;
}
```

➤ Kết quả:

Hello **H**orld!

3. Kỹ thuật xử lý chuỗi

3. Độ dài chuỗi

Để lấy độ dài (số lượng ký tự) của chuỗi ký tự `str`, sử dụng hàm `length()` hoặc hàm `size()`.

Cú pháp: **`str.length()`**; hoặc **`str.size()`**; (với **`str`** là biến kiểu chuỗi)

Ví dụ 9: In ra màn hình số lượng ký tự trong chuỗi "Hello World!".

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str = "Hello World!";
```

```
    cout << "So ky tu trong chuoi la: " << str.length() << " ky tu";
```

```
    return 0;
```

```
}
```

➤ Kết quả: So ky tu trong chuoi la: 12 ky tu

3. Kỹ thuật xử lý chuỗi

4. Rút trích một chuỗi

Trong chuỗi str, rút trích một chuỗi con với chiều dài y bắt đầu tại vị trí x. Nếu không có y, một chuỗi con từ vị trí x đến cuối chuỗi sẽ được rút trích.

Cú pháp: **str.substr(x,y);** (với **str** là biến kiểu chuỗi)

Ví dụ 10: Rút trích chuỗi con từ trong chuỗi "I am learning C++ programming" gồm 12 ký tự, bắt đầu từ ký tự ở vị trí 5.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str = "I am learning C++ programming";
```

```
    cout << "Rut trich chuoi con la: " << str.substr(5, 12);
```

```
    return 0;
```

```
}
```

➤ Kết quả:

Rut trich chuoi con la: learning C++

3. Kỹ thuật xử lý chuỗi

5. Thay thế ký tự trong chuỗi

Trong chuỗi str, thay thế n ký tự tại vị trí bắt đầu là x bằng chuỗi rep_str.

Cú pháp: **str.replace(x, n, rep_str);** (với **str** là biến kiểu chuỗi)

Ví dụ 11: Thay thế chuỗi con "Ky thuat lap trinh" vào trong chuỗi "I am learning C++ programming" tại vị trí ký tự bắt đầu là 14 và số lượng ký tự cần thay thế là 15 ký tự.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str1 = "I am learning C++ programming";
```

```
    string str2 = "Ky thuat lap trinh";
```

```
    str1.replace(14, 15, str2);
```

```
    cout << str1;
```

```
    return 0;
```

```
}
```

➤ Kết quả:

I am learning Ky thuat lap trinh

3. Kỹ thuật xử lý chuỗi

6. Xoá ký tự trong chuỗi

Xoá n ký tự bắt đầu tại vị trí x trong chuỗi ký tự str.

Cú pháp: **str.erase(x,n);** (với **str** là biến kiểu chuỗi)

Ví dụ 12: Xoá 9 ký tự bắt đầu ký tự ở vị trí 5 trong chuỗi "I am learning C++ programming".

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str = "I am learning C++ programming";
```

```
    str.erase(5,9);
```

```
    cout << "Chuoi sau khi xoa la: " << str;
```

```
    return 0;
```

```
}
```

➤ Kết quả:

Chuoi sau khi xoa la: I am C++ programming

3. Kỹ thuật xử lý chuỗi

7. Kiểm tra chuỗi có trong chuỗi

Kiểm tra chuỗi con `sub_str` có xuất hiện trong chuỗi `str` hay không. Nếu có thì trả về vị trí bắt đầu xuất hiện chuỗi `sub_str` trong chuỗi `str`.

Cú pháp: `str.find(sub_str);` (với **str** là biến kiểu chuỗi)

Ví dụ 13: Kiểm tra chuỗi con "lea" có trong chuỗi "I am learning C++ programming" hay không.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str = "I am learning C++ programming";
```

```
    string sub_str = "lea";
```

```
    cout << "Ket quakiem tra la: " << str.find(sub_str);
```

```
    return 0;
```

```
}
```

➤ Kết quả:

Ket quakiem tra la: 5

3. Kỹ thuật xử lý chuỗi

8. Hoán đổi nội dung hai chuỗi.

Hoán đổi nội dung hai chuỗi str1 và str2.

Cú pháp: `str1.swap(str2);` (với **str1** và **str2** là biến kiểu chuỗi)

Ví dụ 14: Hoán đổi nội dung của chuỗi "Hello World!" và chuỗi "I am learning C++ programming".

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    string str1 = "Hello World!";
```

```
    string str2 = "I am learning C++ programming";
```

```
    str1.swap(str2);
```

```
    cout << "str1 sau khi hoan doi: " << str1 << endl;
```

```
    cout << "str2 sau khi hoan doi: " << str2 << endl;
```

```
    return 0;
```

```
}
```

➤ Kết quả:

str1 sau khi hoan doi: I am learning C++ programming

str2 sau khi hoan doi: Hello World!

3. Kỹ thuật xử lý chuỗi

9. Thêm chuỗi.

Thêm chuỗi str2 vào str1 sau vị trí index chỉ định.

Cú pháp: `str1.insert(index, str2);` (với **str1** và **str2** là biến kiểu chuỗi)

Ví dụ 15: Thêm chuỗi "I am learning C++ programming" vào "Hello World!" sau vị trí 6.

```
#include <iostream>
using namespace std;
int main() {
    string str1 = "Hello World!";
    string str2 = "I am learning C++ programming ";
    str1.insert(6,str2);
    cout << "str1 sau khi them str2 la: " << str1 << endl;
    return 0;
}
```

➤ Kết quả:

str1 sau khi them str2 la: Hello I am learning C++ programming World!