

TIN ĐẠI CƯƠNG

STRING

Trần Thị Ngân

Bộ môn Công nghệ phần mềm, Khoa CNTT

Trường đại học Thủy Lợi

Nội dung chính

1. Kiểu dữ liệu string
2. Khai báo và sử dụng
3. Các hàm liên quan
4. Bài tập

1. Kiểu dữ liệu string

- Nhu cầu: nhiều phần mềm làm việc với dãy các kí tự, như họ tên học sinh, địa chỉ, email, ...
→ Tất cả đều ở dạng các chuỗi các kí tự
- Chuỗi các kí tự → có thể sử dụng mảng hoặc vector để lưu trữ.

Kiểu dữ liệu string

- Dưới dạng mảng hoặc vector, chuỗi ký tự được khai báo: `char ten_hoc_sinh[100];` hoặc
`vector <char> ten_hoc_sinh ;`
- Do việc sử dụng các chuỗi kí tự ngày càng phổ biến, C++ đã cung cấp kiểu dữ liệu mới string cùng các hàm tiện ích đi kèm

2. Khai báo và sử dụng

- Khai báo thư viện string trước khi sử dụng :

`#include<string>`

- Cách khai báo biến

`string a; // a la kieu string, a khong co ki tu nao`

`string b = "Dai hoc Thuy Loi";`

`string c("Xin chao");`

- Lưu ý :

Giá trị của biến string phải để trong hai dấu nháy kép " "

Ví dụ: nếu khai báo `string b = Dai hoc Thuy Loi;`

→ thiếu dấu " ", trình biên dịch sẽ báo lỗi

Nếu muốn có dấu nháy kép " " trong string thì cần thêm kí tự \ trước kí tự này

Ví dụ : `string d = "Doremon noi: \" Nobita oi. \" ";`

Khai báo và sử dụng

- In string ra màn hình: sử dụng **cout**

```
string ho_va_ten = "Nguyen Ngoc Minh";  
cout << ho_va_ten;
```

- Nhập string từ bàn phím : 2 cách

- Sử dụng **cin** : chỉ lấy các kí tự đầu đến khi có khoảng trống thì dừng lại

```
string ho_va_ten;  
cin >> ho_va_ten;
```

- Sử dụng **getline**: lấy toàn bộ dòng

```
string ho_va_ten;  
getline(cin, ho_va_ten);
```

Ví dụ

Viết chương trình nhập vào tên của bạn và hiển thị ra màn hình lời chào.

Chỉ số (index)

- Về bản chất, string là 1 chuỗi các kí tự
 - có thể truy cập từng kí tự bằng cách sử dụng chỉ số (index)
- Ví dụ :

```
string a = "Nguyen Anh";
```

→ `a[0] = N, a[4] = e`

```
cout << a[7] << endl; //in ra 'A'
```

```
a[9] = 'g'; //đổi 'h' thành 'g'
```

	N	g	u	y	e	n		A	n	h
Chỉ số	0	1	2	3	4	5	6	7	8	9

3. Các hàm liên quan

- Rất nhiều hàm trong thư viện string, tham khảo <http://www.cplusplus.com/reference/string/string/>
- Cần thực hành thật nhiều để sử dụng thông thạo các hàm
- Một số hàm hay sử dụng

`s.size()` hoặc `s.length()` : trả về chiều dài (số kí tự) của xâu

`s.resize(m)` : thay đổi cỡ của xâu s thành m phần tử

`s.erase(pos,k)` : xoá k kí tự bắt đầu từ vị trí pos

`s.push_back(c)` : thêm kí tự c vào cuối xâu

`s.insert(pos,str)` : chèn xâu str vào s bắt đầu từ vị trí pos

`s.find(str)`: trả về vị trí của str trong s, hoặc `s.npos` nếu không tìm thấy

`s.substr(pos,k)`: trả về xâu mới là k kí tự của s bắt đầu từ vị trí pos

Ví dụ : Thực hiện các
được nhập từ bàn phím.

Ví dụ

Viết chương trình nhập vào tên của bạn và hiển thị ra màn hình lời chào.

Một số toán tử

- `s[i]` : truy cập đến phần tử thứ `i` của chuỗi
`string ho = "Nguyen";` → `ho[2] = 'u'`
- Toán tử `+` : nối hai chuỗi
`string ho = "Nguyen"; string ten = "Ngoc Minh";`
`string hoten = ho + " " + ten;`
→ `hoten = "Nguyen Ngoc Minh"`
- Toán tử `+=` : nối hai chuỗi
`string ho = "Nguyen"; string ten = "Ngoc Minh";`
`string ho += " " + ten;` → `ho = "Nguyen Ngoc Minh"`
- Các phép toán **so sánh** (so sánh theo giá trị từ điển)
`s1 < s2`, `s1 <= s2`, `s1 == s2`, `s1 != s2...`

Một số hàm kí tự

Một số hàm kí tự trong thư viện ctype

int **isalnum**(char c): trả về true nếu c là một chữ cái hoặc một số

int **isalpha**(char c): trả về true nếu c là một chữ cái

int **isdigit**(char c): trả về true nếu c là một chữ số

int **isupper**(char c): trả về true nếu c là một chữ hoa

int **isspace**(char c): trả về true nếu c là một khoảng trống

int **toupper**(char c): đổi kí tự c sang chữ hoa

int **tolower**(char c): đổi kí tự c sang chữ thường

Ví dụ : Thực hiện các hàm trên với các kí tự của xâu

s = "Chelsea 5 - 0 Everton".

Ví dụ

Đổi một xâu kí tự thành chữ hoa.

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4 int main ()
5 {
6     string str;//khai bao
7     cout<<"Nhap vao 1 xau: ";
8     getline(cin,str);//nhap xau
9     cout<<"Xau vua nhap: "<<str;//in lai xau
10    for (int i=0;i<str.length();i++)
11        str[i]=toupper (str[i]);//doi chu cai thanh in hoa
12    cout<<"\nXau sau khi doi thanh chu hoa: "<<str;
13 }
```

Ví dụ

Đổi một chuỗi kí tự thành chữ hoa.

```
#include <iostream>
#include <string>
using namespace std;
//ham chuHoa
string chuHoa(string s)
{
    for (int i=0; i<s.size(); i++)
        s[i] = toupper(s[i]);
    return s;
}

int main()
{
    string str;
    cout << "Nhap vao mot chuoai ki tu: ";
    getline(cin, str); //nhan xau tu ban phim
    cout << "Xau vua nhap duoc viet hoa thanh " << chuHoa(str) << endl;
    return 0;
}
```

4. Bài tập

Bài 1: Viết hàm kiểm tra xem một chuỗi s có chứa toàn các chữ số hay không. Nhập vào một chuỗi bất kỳ, hiển thị kết quả kiểm tra đối với chuỗi vừa nhập.

4. Bài tập

Bài 2: Nhập vào một chuỗi kí tự. Đếm xem kí tự 'a' xuất hiện bao nhiêu lần trong chuỗi đó. Đổi toàn bộ các kí tự 'a' trong chuỗi thành 'e'.