

TIN ĐẠI CƯƠNG

Bài 10: STRING

Trần Thị Ngân

Bộ môn Công nghệ phần mềm, Khoa CNTT

Trường đại học Thủy Lợi

So sánh mảng và vector

- Mảng một chiều :

- Dãy các phần tử có cùng kiểu dữ liệu
- Kích thước cố định

Ví dụ : `float a[10];`

- Vector

- Tương tự như mảng nhưng kích thước có thể thay đổi
- Ví dụ : `vector<float> a;`

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

- Sử dụng `cin`

```
string ho_va_ten ;
```

```
cin >> ho_va_ten ;
```

→ nếu người sử dụng nhập "Nguyen Ngoc Minh", kết quả :

```
ho_va_ten = Nguyen
```

- Sử dụng `getline`

```
string ho_va_ten ;
```

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

→ nếu người sử dụng nhập "Nguyen Ngoc Minh", kết quả :

```
ho_va_ten = Nguyen Ngoc Minh
```


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 hàm trên với xâu

s = "Xin chao cac ban !" được nhập từ bàn phím.

Ví dụ

Chương trình sau đây làm những việc gì?

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str;
    cout << "Nhập vào một chuỗi kí tự: ";
    getline(cin, str); //nhận xâu từ bàn phím
    cout << "Chuỗi này có độ dài là " << str.length() << endl;
    cout << str.erase(9,4) << endl;
    cout << str.insert(str.size()-1," Lan") << endl;
    size_t pos = str.find("chao");
    cout << "pos = " << pos << endl;
    if (pos != str.npos)
    {
        cout << "pos = " << pos << endl;
        cout << str.substr(pos) << endl;
    }
    return 0;
}
```

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

Chương trình sau đây cho ra kết quả thế nào?

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str("Chelsea 5 - 0 Everton");
    //kiem tra mot ki tu co phai la chu cai hay so
    if (isalnum(str[10])) cout << "Chu cai/so " << str[10] << endl;
    else cout << "Ki tu " << str[10] << " khong phai chu cai hay so.\n";
    //kiem tra mot ki tu co phai la chu thuong
    if (islower(str[2])) cout << "Ki tu " << str[2] << " la chu thuong.\n";
    else cout << "Ki tu " << str[2] << " khong phai chu thuong.\n";
    //dem so chu so va so khoang trong
    int i, soChuSo=0, soKhoangTrong=0;
    for (i=0; i< str.size(); i++)
    {
        if (isdigit(str[i])) soChuSo++;
        if (isspace(str[i])) soKhoangTrong++;
    }
    cout << "So chu so la " << soChuSo << endl;
    cout << "So khoang trong la " << soKhoangTrong << endl;
    return 0;
}
```

Ví dụ

Đổi một xâu 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: Nhập vào một xâu 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'.

Bài 2: Nhập vào một chuỗi s. Kiểm tra xem s có chứa toàn các chữ số hay không.

Bài 3: Nhập vào hai xâu str1 và str2. Kiểm tra xem xâu str2 có xuất hiện trong xâu str1 hay không, nếu có thì xuất hiện mấy lần.