

# C++ Strings: Phép thuật của Ngôn ngữ

## Hành trình qua 5 Trạm dừng chân

Slide Learning C++

Ngày 14 tháng 1 năm 2026

# Lời chào từ thế giới Lập trình

## String là gì?

Nếu máy tính chỉ biết đến những con số 0 và 1 khô khan ('int', 'float'), thì **String** (Chuỗi ký tự) chính là "phép thuật" giúp máy tính hiểu và nói được ngôn ngữ con người.

- Hãy tưởng tượng string giống như một **xâu chuỗi hạt**.
- Mỗi chữ cái ('char') là một hạt cườm lấp lánh trên đó.

# Bản đồ kho báu String (Lộ trình học tập)

Chúng ta sẽ đi qua 5 trạm dừng chân thú vị:

- ① **Trạm 1:** Tạo tác Xâu Chuỗi (C++ Strings Basic)
- ② **Trạm 2:** Phép thuật Nối Dây (Concatenation)
- ③ **Trạm 3:** Thước đo thần kỳ (Length)
- ④ **Trạm 4:** Kỹ thuật "Gắp Hạt" & "Đổi Màu" (Access & Change)
- ⑤ **Trạm 5:** Chiếc Hộp Thần Kỳ (User Input)

# Trạm 1: Tạo tác Xâu Chuỗi

## Bức tranh toàn cảnh: Hộp Dựng Chữ

Bộ nhớ máy tính giống như một cái tủ đựng đồ:

- `int`: Ngăn đựng số (5, 10, 100).
- `string`: Ngăn đựng văn bản (Tên, câu chuyện, lời bài hát).

## Hộp công cụ bí mật

Để dùng `string`, bắt buộc phải khai báo thư viện: `#include <string>`

## Quy tắc vàng

Giá trị của chuỗi phải luôn nằm trong **dấu ngoặc kép ("")**.

- Đúng: "Xin chao"
- Sai: 'Xin chao' (Dấu nháy đơn chỉ dành cho char).

```
1 #include <iostream>
2 #include <string> // Hop cong cu can thiet
3 using namespace std;
4
5 int main() {
6     // Tao mot bien string
7     string loiChao = "Xin chao! To la lap trinh vien.";
8
9     // In chuoi ra man hinh
10    cout << loiChao;
11    return 0;
12 }
13
```

# Kiểm tra nhanh (Check-up 1)

**Câu hỏi:** Nếu viết code thế này thì máy tính có hiểu không? Tại sao?

```
1 string tenCuaBan = 'Nam';  
2
```

# Kiểm tra nhanh (Check-up 1)

**Câu hỏi:** Nếu viết code thế này thì máy tính có hiểu không? Tại sao?

```
1 string tenCuaBan = 'Nam';
2
```

**Đáp án: Báo lỗi!**

Máy tính sẽ "dối" ngay lập tức.

- Dấu nháy đơn '' chỉ dành cho một hạt cườm (char).
- Dấu nháy kép "" mới dành cho cả xâu chuỗi (string).

## Trạm 2: Phép thuật Nối Dây

Hãy tưởng tượng mỗi biến string là một **toa tàu**.

### Dấu Cộng (+) thần thánh

Trong thế giới String, dấu + không phải là cộng giá trị toán học, mà là **dán dính** các toa tàu lại.

Ví dụ: "Nguyen" + "An" = "NguyenAn"

**Lưu ý:** Máy tính không tự thêm dấu cách. Ta phải tự chèn một "toa tàu rỗng" vào giữa.

# Ví dụ Nối chuỗi

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string ho = "Nguyen";
7     string ten = "An";
8
9     // Cach 1: Dinh chum (Xau)
10    // string hoVaTen = ho + ten; -> "NguyenAn"
11
12    // Cach 2: Co them dau cach " " (Dep)
13    string hoVaTen = ho + " " + ten;
14
15    cout << hoVaTen; // In ra: Nguyen An
16    return 0;
17}
18
```

## Góc chuyên gia

Có thể dùng hàm append() để nối đuôi, nhưng dùng + sẽ tự nhiên hơn.

# Kiểm tra nhanh (Check-up 2)

**Câu hỏi:** Kết quả của đoạn code sau là gì?

```
1 string x = "10";
2 string y = "20";
3 string z = x + y;
4 cout << z;
5
```

- A. 30
- B. 1020
- C. Báo lỗi

# Kiểm tra nhanh (Check-up 2)

**Câu hỏi:** Kết quả của đoạn code sau là gì?

```
1 string x = "10";
2 string y = "20";
3 string z = x + y;
4 cout << z;
5
```

- A. 30
- B. 1020
- C. Báo lỗi

**Đáp án:** B (1020)

Vì x và y nằm trong dấu ngoặc kép, máy tính coi chúng là **văn bản**, không phải con số. Nó sẽ dán số "10" bên cạnh số "20".

## Trạm 3: Thước đo thần kỳ

Để đếm số lượng ký tự trong chuỗi (ví dụ kiểm tra độ dài mật khẩu), ta dùng hàm `.length()`.

```
1 string bangChuCai = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
2 cout << bangChuCai.length();
3 // Kết quả: 26
4
```

### Bí mật của "Khoảng trắng"

Máy tính đếm tất cả mọi thứ trong dấu ngoặc kép, bao gồm cả **dấu cách (space)**.

Ví dụ: "Xin chao" có độ dài là **8** (chứ không phải 7).

Lưu ý: `.size()` và `.length()` trong C++ String tương đương nhau.

# Kiểm tra nhanh (Check-up 3)

**Câu hỏi:** Đoạn code sau in ra số mấy?

```
1 string biMat = "A B C";
2 cout << biMat.length();
3
```

# Kiểm tra nhanh (Check-up 3)

**Câu hỏi:** Đoạn code sau in ra số mấy?

```
1 string biMat = "A B C";
2 cout << biMat.length();
3
```

Đáp án: 5

Hãy đếm ký:

- ① A
- ② (dấu cách)
- ③ B
- ④ (dấu cách)
- ⑤ C

Tổng cộng là 5 ký tự.

## Trạm 4: Kỹ thuật "Gắp Hạt" & "Đổi Màu"

Biến **string** giống như dãy tủ khóa. Để lấy đồ, bạn cần biết số thứ tự (index).

### Quy tắc Số 0

Máy tính bắt đầu đếm từ số **0**, không phải số 1.

Ví dụ chuỗi "**Hello**":

H	e	I	I	o
<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>

# Thay đổi ký tự

Bạn có thể thay đổi nội dung từng "ngăn tủ".

```
1 string monAn = "Hello";
2
3 // 1. Truy cap (Lay ra)
4 cout << monAn[0]; // In ra: H
5
6 // 2. Thay doi (Doi mau)
7 monAn[0] = 'J';
8
9 cout << monAn; // In ra: Jello
10
```

## Cảnh báo quan trọng

Khi làm việc với từng ký tự đơn lẻ (`monAn[0]`), bắt buộc dùng **đầu nháy đơn** ' '.

- Đúng: `monAn[0] = 'J';`
- Sai: `monAn[0] = "J";`

# Kiểm tra nhanh (Check-up 4)

**Câu hỏi:** Chữ cái nào sẽ hiện lên màn hình?

```
1 string ten = "VIETNAM";
2 cout << ten[4];
3
```

- A. T
- B. N
- C. A

# Kiểm tra nhanh (Check-up 4)

**Câu hỏi:** Chữ cái nào sẽ hiện lên màn hình?

```
1 string ten = "VIETNAM";
2 cout << ten[4];
3
```

- A. T
- B. N
- C. A

**Đáp án:** B (Chữ N)

Đếm từ 0: V(0) - I(1) - E(2) - T(3) - **N(4)**.

# Trạm 5: Chiếc Hộp Thần Kỳ

## Vấn đề của cin

cin giống như người đưa thư lười biếng. Nó dừng lại ngay khi gặp dấu cách. Nếu nhập "Tran Hung", cin chỉ lấy được "Tran".

## Giải pháp: getline()

Dùng "máy hút bụi" getline để lấy cả dòng. Cú pháp: getline(cin, TenBien);

```
1 string hoVaTen;
2 cout << "Nhập tên đầy đủ: ";
3 // cin >> hoVaTen; -> Lỗi nếu có dấu cách
4 getline(cin, hoVaTen); // -> Lấy trọn vẹn cả dòng
5 cout << "Xin chào: " << hoVaTen;
6
```

# Tổng kết hành trình

Chúng ta đã thu được các chiến lợi phẩm:

- ① **Khởi tạo:** #include <string> và dùng "".
- ② **Nối chuỗi:** Dùng dấu + để dính các toa tàu.
- ③ **Độ dài:** Dùng .length() (đếm cả dấu cách).
- ④ **Truy cập:** Dùng [ ] và đếm từ 0.
- ⑤ **Nhập liệu:** Dùng getline(cin, bien) để nhập cả câu.

# Bài tập về nhà (Challenge)

## Đề bài

Viết chương trình thực hiện 3 việc:

- ① Nhập vào **tên đầy đủ** (Ví dụ: "Le Loi").
- ② In ra **chữ cái đầu tiên**.
- ③ In ra **độ dài** của tên.

# Lời giải tham khảo

Đây là đoạn code giải quyết thử thách trên:

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string name;
7
8     cout << "Nhập tên của bạn: ";
9     // Dùng getline để nhập tên có dấu cách
10    getline(cin, name);
11
12    cout << "Chữ cái đầu tiên: " << name[0] << endl;
13
14    // Dùng .size() hoặc .length() đều được
15    cout << "Độ dài tên bạn: " << name.size() << endl;
16
17    return 0;
18}
```

# Thử thách nâng cao (Bonus Stage)

## Đề bài khó

Viết chương trình nhập tên, in ra chữ cái **CUỐI CÙNG** trong tên.

### Gợi ý tư duy:

- Tên "NAM"(độ dài 3), chữ cuối 'M' ở vị trí số 2.
- Tên "TUAN"(độ dài 4), chữ cuối 'N' ở vị trí số 3.
- Quy luật: Vị trí cuối = Độ dài - 1

*Chúc các bạn thành công trên con đường lập trình C++!*