

## Bài thực hành 13

### Mục tiêu

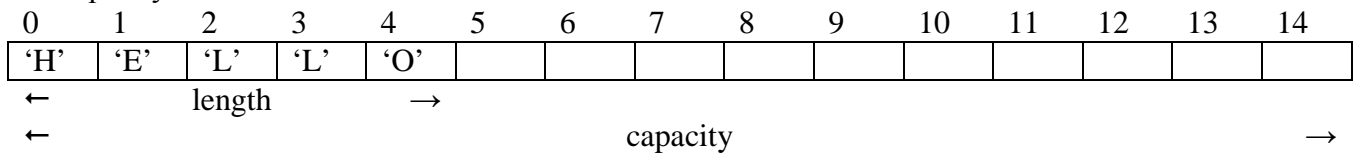
- Ôn tập: con trỏ và cấp phát động

### Bài tập

**Bắt buộc giải phóng tất cả các biến được cấp phát động khi không còn cần đến chúng.**

Bài này yêu cầu cài đặt lớp StringClass biểu diễn chuỗi ký tự, liên hệ mật thiết với các kiến thức trong phần 10.3 giáo trình. Bạn không được sử dụng các hàm xử lý chuỗi ký tự có sẵn trong <cstring> hay <string>.

Để biểu diễn chuỗi ký tự bằng mảng char cấp phát động, lớp có biến thành viên private tên là data kiểu char \*. Ngoài ra còn có biến thành viên length kiểu int để đếm số lượng ký tự trong chuỗi, biến thành viên capacity kiểu int lưu kích thước của mảng data. Bên dưới là hình vẽ minh họa mảng data, length và capacity khi lưu chuỗi "HELLO".



Tệp StringClass.h chứa định nghĩa của lớp StringClass được cho sẵn.

Hãy cài đặt các hàm thành viên và đặt vào StringClass.cpp và viết hàm main (tệp main.cpp) chạy thử các hàm thành viên này thông qua các đối tượng StringClass.

#### StringClass.h

```
#ifndef STRING_CLASS_H
#define STRING_CLASS_H
#include <iostream>
using namespace std;

class StringClass{
public:
    StringClass();
    StringClass(char * cstr); // Tao bien StringClass moi co du lieu cua xau cstr
    StringClass(const StringClass& sc);
    ~StringClass();

    int getLength()const; // Tra ve so ki tu trong xau
    int getCapacity()const; // Tra ve dung luong cap phat cho mang data
    void setCapacity(int newCapacity); // Thay doi dung luong toi da cua xau

    bool isEmpty()const; // Kiem tra xem xau co rong hay khong
    void clear(); // Xoa noi dung cua xau

    char& operator[](int i); // Tra ve tham chieu den ki tu co chi so i
    StringClass& operator=(const StringClass& sc); // Nap chong phep gan
    StringClass& operator+=(const StringClass& sc); // Nap chong phep +=
    const StringClass operator+(const StringClass& sc); // Nap chong phep +

    friend istream& operator>>(istream& in, StringClass& sc);
    friend ostream& operator<<(ostream& out, const StringClass& sc);
};
```

```
private:
    char * data;
    int length;
    int capacity;
};

#endif
```