

Trạng thái	Đã xong
Bắt đầu vào lúc	Chủ Nhật, 30 tháng 3 2025, 10:10 PM
Kết thúc lúc	Chủ Nhật, 30 tháng 3 2025, 10:20 PM
Thời gian thực hiện	9 phút 37 giây

Câu hỏi 1

Đúng

Viết hàm void void reverse(char str[]) đảo ngược các ký tự trong chuỗi.

- Mô tả hàm: Hàm này thực hiện việc đảo ngược chuỗi ký tự truyền vào ngay tại chỗ (in-place). Hàm không trả về giá trị và thay đổi chuỗi đầu vào để chuỗi sau khi đảo ngược thay thế chuỗi ban đầu.

- Đầu vào: char str[]

Mảng ký tự chứa chuỗi cần đảo ngược

- Đầu ra:

Chuỗi đầu vào được đảo ngược ngay tại chỗ, không cần chuỗi đầu ra outstr.

- Các thư viện có thể dùng: <iostream>, <cstring>

- Note: Sinh viên không được sử dụng các từ khoá sau trong đoạn code nộp bài, kể cả trong comment: "include", "string"
Tham số truyền vào:

Implement void reverse(char str[]) function which reverses the string.

- Function Description: This function reverses the input character string in-place. It does not return a value and modifies the input string so that the reversed string replaces the original string.

- Input: char str[]

Char array contains string

- Output:

The input string is reversed in place, no need for an output string outstr.

- Available library: <iostream>, <cstring>

- Note: Your code MUST NOT contain any of the words in the list {"include", "string"} even in the comment

For example:

Test	Result
const int MAX_SIZE = 100; char str[] = "this is my school"; reverse(str); cout << str;	loohcs ym si siht

Answer: (penalty regime: 0 %)

Reset answer

```
1 // Đảo ngược chuỗi ký tự
2 void reverse(char str[]) {
3     // Tìm độ dài chuỗi
4     int length = 0;
5     while (str[length] != '\0') {
6         length++;
7     }
8
9     // Đảo ngược chuỗi
10    for (int i = 0; i < length / 2; i++) {
11        char temp = str[i];
12        str[i] = str[length - 1 - i];
13        str[length - 1 - i] = temp;
14    }
15 }
```

	Test	Expected	Got	
✓	<pre>const int MAX_SIZE = 100; char str[] = "this is my school"; reverse(str); cout << str;</pre>	loohcs ym si siht	loohcs ym si siht	✓

Passed all tests! ✓

Câu hỏi 2

Đúng

Một đường truyền thông tin gồm các kí tự bị hacker tấn công. Cuộc tấn công chỉ làm ảnh hưởng đến các tín hiệu có giá trị là kí tự trong bảng chữ cái làm cho các chữ thường thành chữ in hoa và ngược lại các chữ in hoa thành chữ thường. Sinh viên được chủ đường truyền nhờ để khôi phục dữ liệu lại về như ban đầu.

Sinh viên viết hàm xử lý `void recover(char signal[])` để biến đổi tất cả chữ cái in hoa thành chữ viết thường và chữ cái thường thành chữ in hoa có trong chuỗi `signal` (lưu ý biến đổi trực tiếp trên `signal`)

Đầu vào (Input):

- Chuỗi kí tự cần biến đổi chữ hoa thành chữ thường và chữ thường thành chữ hoa

Đầu ra (Output):

- Kết quả biến đổi trực tiếp trên `signal` (lưu ý các kí tự không phải kí tự trong bảng chữ cái thì giữ nguyên)

The information transmission line is attacked by hackers. The attack only affects the signals with characters that have a value in the alphabet, converting lowercase letters to uppercase and vice versa. The student is asked by the transmission owner to restore the data to its original state.

The student writes a processing function `void recover(char signal[])` to transform all uppercase letters into lowercase letters and lowercase letters into uppercase letters in the string `signal` (note the transformation is inplace).

Input:

- A character string to transform uppercase letters to lowercase and lowercase letters to uppercase

Output:

- The transformed result in `signal`

For example:

Test	Result
char signal[] = "hello world" ; recover(signal); cout << signal;	HELLO WORLD
char signal[] = "CSE HCMUT" ; recover(signal); cout << signal;	cse hcmut
char signal[] = "I waNT tO hAVe 10 MaRk In FudaMEnTAL of PRoGrAmInG couRsE." ; recover(signal); cout << signal;	i wAnt To Have 10 mArK iN FUDaMEnTAl OF prOGraMiNg COUrSe.

Answer: (penalty regime: 0 %)

Reset answer

```

1 // Hàm khôi phục tín hiệu bị tấn công
2 void recover(char signal[]) {
3     int i = 0;
4
5     // Duyệt qua từng ký tự trong chuỗi
6     while (signal[i] != '\0') {
7         // Kiểm tra nếu là chữ in hoa (A-Z)
8         if (signal[i] >= 'A' && signal[i] <= 'Z') {
9             // Chuyển in hoa thành chữ thường
10            signal[i] = signal[i] + ('a' - 'A');
11        }
12        // Kiểm tra nếu là chữ thường (a-z)
13        else if (signal[i] >= 'a' && signal[i] <= 'z') {
14            // Chuyển chữ thường thành in hoa
15            signal[i] = signal[i] - ('a' - 'A');
16        }
17        // Các ký tự khác giữ nguyên
18
19        i++; // Chuyển đến ký tự tiếp theo
20    }
21 }
```



^