

Trạng thái	Đã xong
Bắt đầu vào lúc	Thứ Tư, 16 tháng 4 2025, 4:26 PM
Kết thúc lúc	Thứ Tư, 16 tháng 4 2025, 4:32 PM
Thời gian thực hiện	5 phút 44 giây

Câu hỏi 1

Đúng

Đạt điểm 1,00

Viết hàm void process(const char str[], char outstr[]) chuẩn hóa tên người (tên tiếng Anh). Quy tắc chuẩn hóa như sau:

1. Chữ cái đầu tiên của mỗi từ phải ghi hoa, các chữ cái còn lại ghi thường.
2. Tên người chỉ bao gồm những ký tự trong bảng chữ cái alphabet tiếng Anh.
3. Không có các khoảng trắng thừa. Cụ thể, không có 2 khoảng trắng liền kề nhau và không có khoảng trắng nào ở đầu và cuối chuỗi.

- Đầu vào:

Mảng ký tự chứa chuỗi cần chuẩn hóa

- Đầu ra:

Tên người đã được chuẩn hóa được ghi vào chuỗi outstr

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

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

Implement void process(const char str[], char outstr[]) function which standardizes English name. The standardized name has the following features:

1. The first character of each word must be uppercase. The remaining characters of each word must be lowercase.
2. Only contains characters in English alphabet.
3. There are no consecutive space characters and there is no space character at the beginning/end of the name

- Input:

Char array contains string

- Output:

The standardized name is written to outstr

- Available library: <iostream>, <string>

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

For example:

Test	Result
<pre>const int MAX_SIZE = 100; char str[] = "doAn VAN *(()hau @!#\$" ; char outstr[MAX_SIZE]; process(str, outstr); cout << outstr;</pre>	Doan Van Hau

Answer: (penalty regime: 0 %)

Reset answer

```
1 void process(const char str[], char outstr[]) {
2     int i = 0, j = 0;
3     bool newWord = true;
4
5     // Bỏ qua khoảng trắng ở đầu
6     while (str[i] == ' ') {
7         i++;
8     }
9
10    // Xử lý các ký tự trong chuỗi
11    while (str[i] != '\0') {
```

```

12 // Chỉ xử lý các ký tự chữ cái
13 if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z')) {
14     if (newWord) {
15         // Chữ cái đầu tiên của từ viết hoa
16         if (str[i] >= 'a' && str[i] <= 'z') {
17             outstr[j++] = str[i] - 'a' + 'A';
18         } else {
19             outstr[j++] = str[i];
20         }
21         newWord = false;
22     } else {
23         // Các chữ cái còn lại viết thường
24         if (str[i] >= 'A' && str[i] <= 'Z') {
25             outstr[j++] = str[i] - 'A' + 'a';
26         } else {
27             outstr[j++] = str[i];
28         }
29     }
30 }
31 // Xử lý khoảng trắng
32 else if (str[i] == ' ') {
33     // Thêm khoảng trắng nếu không phải khoảng trắng liền kề
34     if (j > 0 && outstr[j-1] != ' ') {
35         outstr[j++] = ' ';
36         newWord = true;
37     }
38 }
39
40 i++;
41 }
42
43 // Loại bỏ khoảng trắng ở cuối nếu có
44 if (j > 0 && outstr[j-1] == ' ') {
45     j--;
46 }
47
48 // Kết thúc chuỗi
49 outstr[j] = '\0';
50 }

```



	Test	Expected	Got	
✓	<pre>const int MAX_SIZE = 100; char str[] = "doAn VAN *((()hau @!#\$" ; char outstr[MAX_SIZE]; process(str, outstr); cout << outstr;</pre>	Doan Van Hau	Doan Van Hau	✓

Passed all tests! ✓