

1. Yêu cầu:

Khi làm bài tập đặt câu với các vị thần Hy Lạp, Hercules đã vô tình viết sai tên của cha mình là thần Zeus thành "Zues". Hãy viết chương trình giúp Hercules sửa lại những chỗ viết sai này.

Input:

- Một chuỗi kí tự gồm các từ cách nhau bởi dấu cách được viết trên một dòng.

Output:

- In chuỗi kí tự viết đúng tên thần Zeus.

Gợi ý:

1. Đọc chuỗi kí tự sử dụng hàm `getline` và lưu vào biến `string text;`
2. Sử dụng vòng lặp với `i=0` đến `text.size()-1`, in ra từng kí tự `text[i]` nếu `i > text.size()-4` hoặc `text.substr(i,4) != "Zues"`, ngược lại, in ra "Zeus" và nhảy cách đến vị trí `i+4`.

Có thể tra google để biết về hàm `substring` và `size` để xử lý string.

2. Yêu cầu:

Pig Latin là một trò chơi ngôn ngữ trong đó các từ tiếng Anh bị thay đổi, thường là bằng cách thêm một hậu tố giả hoặc bằng cách di chuyển phần bắt đầu hoặc phụ âm đầu tiên hoặc cụm phụ âm của một từ về cuối từ và thêm một âm tiết để tạo ra một hậu tố. Mục đích là để che dấu các từ khỏi những người không biết luật.

Hãy viết hàm `string pigLatin(string word)` để chuyển đổi một từ tiếng Anh đầu vào thành một từ trong ngôn ngữ Pig Latin với các quy tắc sau:

- Nếu từ bắt đầu bởi một phụ âm, chuyển phụ âm xuống cuối của từ và thêm "ay" vào cuối từ. Theo đó, "ball" trở thành "allbay".
- Nếu từ bắt đầu bởi một nguyên âm, thêm "way" vào cuối từ. Theo đó, "one" trở thành "oneway".

Bạn có thể giả sử từ đầu vào là một từ viết thường và không chứa các kí tự đặc biệt.

3. Yêu cầu:

Viết hàm `long long int factorial(int n)` nhận đầu vào là số nguyên dương nn , đầu ra là giá trị $n!=1\times2\times3\times\dots\times nn!=1\times2\times3\times\dots\times n$ hàm trả về `long long` nhưng mà trả về `int` vẫn đúng 1 số testcase

Input:

- Số nguyên dương nn

Output:

- Kết quả giai thừa của nn

Gợi ý:

1. Khai báo biến lưu giá trị giai thừa, có giá trị là 1: `long long int result = 1;`
2. Để tính giai thừa dùng vòng for biến chạy i từ 2 đến n : `result = result * i;`
3. Trả về kết quả

4. Yêu cầu:

Sử dụng switch-case viết hàm `double calculate (double num1, char operat, double num2)` giả lập một máy tính bỏ túi.

Input:

- Tham số `num1` và `num2` tương ứng với 2 số hạng của phép toán
- Tham số `operat` tương ứng với toán tử. Biết các toán tử đơn giản thực hiện phép toán lần lượt là:
 - Phép trừ `--`
 - Phép cộng `++`
 - Phép chia `//`
 - Phép nhân `**`

Output:

- Kết quả của phép toán giữa số `num1` và `num2` với toán tử `operat`
- Kết quả làm tròn đến chữ số thập phân thứ hai.

5. Yêu cầu:

Viết hàm `void readArray(int a[], int n)` để đọc vào nn số nguyên từ bàn phím và lưu vào mảng số nguyên aa và hàm `void printArray(int a[], int n)` để in các phần tử của mảng aa có số lượng phần tử là nn, các phần tử cách nhau bởi dấu cách.

Input:

- Một dãy số nguyên n phần tử.

Output:

- In ra dãy số nguyên đã nhập từ bàn phím
-

6. Vector:

Sử dụng vector để làm lại bài 5.

Link tham khảo kiến thức vector:

<https://www.cplusplus.com/reference/vector/vector>

Hoặc:

<https://vncoder.vn/bai-hoc/vector-khai-niem-257>