<https://skillsmart.ru/algo/lvl1/f27e.html>

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | |  |  |  | | --- | |  | | |  |  | | --- | --- | | |  | | --- | | **Гарри Поттер идёт в первый класс Хогвартса**  *Казалось бы, чего тут сложного -- выучил несколько колдовских слов, произнёс, и... никакого эффекта. Именно поэтому в специальных учебных заведениях наподобие Хогвартса вербальную магию изучают годами. Дело в том, что в волшебных учебниках написаны слова заклинания, которые сами по себе не работают. Их надо уметь преобразовывать в действующую магию с помощью специальных алгоритмов. Давайте познакомимся с одним из них.*  https://skillsmart.ru/algo/lvl1/xg.jpg  У нас имеется не-магическое слово -- стандартная последовательность русских и английских букв (строковый тип). Такие слова в любом языке программирования можно сравнивать, используя обычный лексикографический порядок, когда "а" меньше чем "я", "a" меньше чем "z" и т. д.  Задача: преобразовать такое слово в магическое с помощью любого числа перестановок двух любых букв в слове, либо убедиться, что получить магическое слово невозможно.  Итоговое слово должно отвечать двум требованиям: 1) оно лексикографически больше, чем исходное слово; 2) оно наименьшее из всех лексикографически больших слов, которые можно получить перестановкой пары букв.  Тестовые примеры:  "ая" преобразовываем в "яа"  "fff" - невозможно преобразовать  "нклм" в "нкмл"  "вибк" в "викб"  "вкиб" в "ибвк"  Функция  string BiggerGreater(string input)  получает на вход исходную строку длиной 2 или более русских или английских строковых (маленьких) букв, и возвращает итоговое магическое слово. Если получить его невозможно, возвращается пустая строка. Например, BiggerGreater("вибк") = "викб"; BiggerGreater("fff") = "".  [как постить решение](https://skillsmart.ru/algo/lvl1/how.html) | | | |  | | --- | |  | |