

## Контрольная работа. Строки. 8Д.

В каждой из предложенных ниже задач, нужно реализовать те функции, которые описаны в условии. Функции должны называться, иметь точно такие же аргументы и возвращать те же значения, как написано в условии. В противном случае, система проверки не зачтет ваше решение. Прототипы функций, которые нужно реализовать, писать **НЕ** надо. Система сама вставит их. Также **НЕ** надо писать `main()`.

Если для реализации нужных функций вам понадобятся вспомогательные, используйте их. Для них заголовки писать можно и нужно.

К каждому решению система автоматически подключит библиотеки `stdio.h`, `iostream` и `string.h`. Если вам понадобятся другие библиотеки, подключайте их самостоятельно. Не стоит подключать `TXLib.h`, система вас не поймет.

Во всех задачах, строка может содержать только символы из ASCII таблицы с номерами от 32 до 127 (и, конечно, строка заканчивается нуль-терминалом).

### A. Палиндром.

Реализуйте функцию

```
bool IsPalindrome(const char str[]),
```

которая вернет `true`, если строка является палиндромом, и `false` в противном случае. В данной задаче заглавные и строчные буквы считаются различными. То есть строка "abba" палиндромом является, а строка "Abba" — нет.

### B. Палиндром без учета регистра.

Реализуйте функцию

```
bool IsNonCaseSensitivePalindrome(const char str[]),
```

которая вернет `true`, если строка является палиндромом, и `false` в противном случае. В данной задаче заглавные и строчные буквы считаются одинаковыми. То есть строка "Abba" палиндромом является.

### C. Палиндром без учета лишних символов.

Реализуйте функцию

```
bool IsPalindromeWithGarbage(const char str[]),
```

которая вернет `true`, если строка является палиндромом, если в ней оставить только буквы, и `false` в противном случае. В данной задаче заглавные и строчные буквы считаются различными. Например, строка "A ro?zA upala,,,, na lapu AzorA." является палиндромом в контексте данной задачи.

### D. Форматирование строки

Дана текстовая строка, содержащая буквы латинского алфавита, пробелы, запятые и точки.

Отформатируйте этот текст по следующим правилам:

- В начале и конце строки не должно быть пробелов.
- Все слова разделяются ровно одним пробелом.
- Точки и запятые пишутся слитно с предыдущим словом, после знака препинания ставится пробел (если этот знак препинания не завершает строку).

Выведите полученную строку. После неё обязательно следовать символ перевода строки.

Гарантируется, что никакие два знака препинания не идут подряд.

В этой задаче необходимо сдавать программу целиком, а не отдельную функцию.

### E. Следующий палиндром

Рассмотрим все натуральные числа, запись которых в десятичной системе счисления является палиндромом (при этом запись не начинается с нуля). Например, числа 121 и 1331 являются палиндромами, а число 123 — нет. Напишите функцию

```
void NextPalindrome(const char number[], char next_palindrom[]),
```

которая данному натуральному числу `number` определит следующее за ним натуральное число (то есть наименьшее число, которое превосходит `number`), являющееся палиндромом.

Функция получает на вход строку, состоящую не более чем из 200 цифр. Результат нужно записать в строку `next_palindrom`.