

## СЕМИНАР 25 СЕНТЯБРЯ

1. Напишите следующие функции:

```
// вычисляет длину C-строки str
int strlen (const char * str);

// приписывает вторую C-строку к первой
char * strcat (char * destination, const char * source);

// переворачивает C-строку source
char * strrev (char * destination, const char * source);

// сравнивает две C-строки
int strcmp (const char * str1, const char * str2);
```

Постарайтесь написать как можно более выразительный код.

2. Напишите реализацию двумерного массива чисел, используя класс `vector`. Выведите его «змейкой» – каждую вторую строку от конца к началу.
3. Даны последовательности чисел. Каждая последовательность заканчивается числом 0. Надо загрузить эти последовательности в память, отсортировать каждую из них, а затем вывести эти отсортированные последовательности в «почти лексикографическом» порядке. Но только если одна последовательность короче другой и точно совпадает с началом другой, то сначала должна идти длинная, а не короткая.
4. Даны строки разной длины, состоящие только из цифр. Нужно загрузить строки в память и отсортировать в числовом порядке. (Примечание: преобразовывать строки в числа с помощью функций `atoi` и аналогичных нельзя.)
5. Напишите функцию `template <typename T> void printBinary(T number)`, которая бы выводила на экран битовое представление числа. Протестируйте корректность ее работы для типов `char`, `int`, `long`.
6. Напишите функцию поиска подстроки в строке. Совпадением считается подстрока, совпадающая с образцом с точностью до перестановки букв.