

## Задачи на строки:

1. **ASCII:** Вывести на экран все символы таблицы ASCII. Символы выводятся с помощью функции `printf` с опцией `"%c"`
2. **Модификация строки:** Объявите строку которая может содержать до 50 символов. Проинициализировать её как `"a bunch of characters"`. Отдельными командами изменить строку на `"a Bunch of Characters"`
3. **Усечение строки:** На вход подаётся строка содержащая один или несколько пробелов. Вывести на экран содержимое строки до первого пробела, используя только один вызов функции `printf()`. Строки выводятся/считываются с помощью функции `printf()/scanf()` с опцией `"%s"`
4. **for\_6:** Решить задачу `for_6`
5. **Обратная строка:** Считать строку, "перевернуть" и вывести на экран. То есть если на вход поступает строка `"example"` то вывести нужно `"elpmaxe"`.
6. **Поиск подстроки:** Считать 2 строки и проверить является ли вторая строка подстрокой первой строки. Нужно использовать функцию `char* strstr(const char* str, const char* substr)`. Вывести на экран YES или NO соответственно.
7. **Uppercase:** Написать функцию, которая будет поданную на вход строку переводить из lowercase в UPPERCASE.
8. **for\_14:** Решить задачу `for_14` . (Подсказка: Введите переменную-счётчик, которая будет увеличиваться, когда встречается открывающая скобка, и уменьшаться, когда встречается закрывающая)

## Задачи на строки:

1. **ASCII:** Вывести на экран все символы таблицы ASCII. Символы выводятся с помощью функции `printf` с опцией `"%c"`
2. **Модификация строки:** Объявите строку которая может содержать до 50 символов. Проинициализировать её как `"a bunch of characters"`. Отдельными командами изменить строку на `"a Bunch of Characters"`
3. **Усечение строки:** На вход подаётся строка содержащая один или несколько пробелов. Вывести на экран содержимое строки до первого пробела, используя только один вызов функции `printf()`. Строки выводятся/считываются с помощью функции `printf()/scanf()` с опцией `"%s"`
4. **for\_6:** Решить задачу `for_6`
5. **Обратная строка:** Считать строку, "перевернуть" и вывести на экран. То есть если на вход поступает строка `"example"` то вывести нужно `"elpmaxe"`.
6. **Поиск подстроки:** Считать 2 строки и проверить является ли вторая строка подстрокой первой строки. Нужно использовать функцию `char* strstr(const char* str, const char* substr)`. Вывести на экран YES или NO соответственно.
7. **Uppercase:** Написать функцию, которая будет поданную на вход строку переводить из lowercase в UPPERCASE.
8. **for\_14:** Решить задачу `for_14` . (Подсказка: Введите переменную-счётчик, которая будет увеличиваться, когда встречается открывающая скобка, и уменьшаться, когда встречается закрывающая)