**ЛАБОРАТОРНАЯ РАБОТА**

**Лабораторная работа №13**

**Работа со строками**

**Вариант №6**

по дисциплине

«Алгоритмизация и программирование»

A picture containing dark, night, night sky, hydrozoan

Description automatically generated

Выполнил студент гр.3530904/20003 Козак Б.П.

Оглавление

[Задача №1 3](#_Toc122104748)

[Требования 5](#_Toc122104749)

[Приложение 1 6](#_Toc122104755)

[Приложение 2 7](#_Toc122104756)

# Задача №1

1. Для работы со строками создать свой класс String.  
   Считать, что один символ – один байт.
2. Класс String должен содержать:
   1. private:
      1. size – длина строки, тип std::size\_t (#include <cstddef>)
      2. capacity – размер памяти, выделенный под строку, тип std::size\_t
      3. pointer – адрес начала строки, тип char\*
   2. public
      1. String() - Конструктор, создающий “пустую” строку.
      2. String(const char\* str) - Конструктор, создающий строку на основе строки в стиле С (с завершающим нулем).
      3. String(const String& str) - Конструктор копирования.
      4. String(String&& str) noexcept - Конструктор перемещения.
      5. ~String () - Деструктор.
      6. String& operator= (const char\* str) - Оператор присваивания строки в стиле С (с завершающим нулем).
      7. String& operator= (String&& str) noexcept - Оператор перемещения.
      8. char& operator[] (std::size\_t pos) - Возвращает символ находящийся по индексу в строке начиная с 0 до size() – 1.
      9. String& insert (std::size\_t pos, const char\* str) - Вставка строки в стиле С в строку типа String в заданную позицию pos.
      10. int compare (const String& str) - Cравнение двух строк типа String. Возвращает число <0, если вызывающая строка лексикографически меньше str, число > 0, если больше, число = 0, если строки равны
3. Расширьте возможности класса String
   1. Методы:
      1. String& append (const char\* s);
      2. size\_t rfind (char c, size\_t pos = npos) const;
      3. void clear();
   2. Сортировка массива строк по методу «поплавка».
   3. Определите:
      1. есть ли среди введенных строк строки, содержащие заданный символ;
      2. среди строк, содержащих заданный символ, определите строку максимальной длины

# Требования

### 1. Размер массива должен быть задан корректно. 1.1 Размер массива, заданный пользователем, должен быть задан корректно.

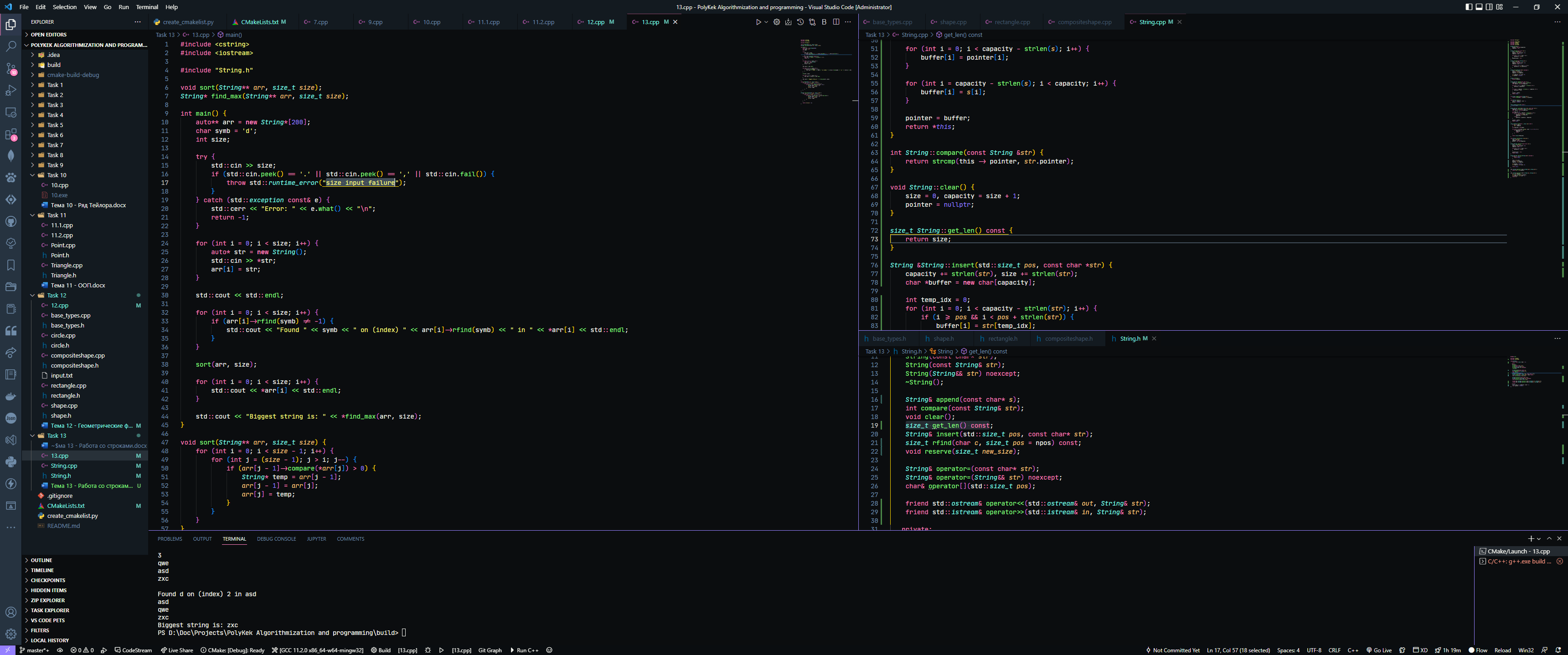
### Размер массива, заданный пользователем, должен быть задан целым числом. Если размер задан неверно, то программа выведет “ERROR: size input failure” и завершится с ошибкой.

## Таблица с детальными требованиями и тест планом

|  |  |  |  |
| --- | --- | --- | --- |
| Требование | Детальные требования | Данные | Ожидаемый результат |
| 1. Размер массива, заданный пользователем, должен быть задан корректно | | | |
| * 1. Файл, путь к которому задан пользователем, должен открываться | Размер массива, заданный пользователем, должен быть задан целым числом. Если размер задан неверно, то программа выведет “ERROR: size input failure” и завершится с ошибкой |  | Сообщение «ERROR: size input failure». |

## 

## Приложение 1



(рис 1)

A screenshot of a computer

Description automatically generated with medium confidence

(рис 2)

## Приложение 2

<https://github.com/OverFitted/polytech-AIP-course>

Репозиторий проекта с кодом программы