МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ автономное ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

(ФГБОУ ВПО МПУ)



Кафедра СМАРТ-технологии

Лабораторная работа № 1

“Изучение понятия алгоритма и основных видов алгоритмов”

По дисциплине «Программирование и основы алгоритмизации систем управления»

Группа \_\_\_\_\_\_\_\_\_\_241-324\_\_\_\_

№ группы

Студент Сальников Лев Владимирович

Подпись студента

Дата \_\_\_\_\_\_\_\_\_\_\_29.11.2024\_\_\_\_\_\_\_\_\_\_

Дата сдачи

Преподаватель Клецкин А. А.

Байрамов Э. В.

Подпись преподавателя

2024

Цель:

Ознакомиться с понятием алгоритма, изучить различные виды алгоритмов и их применение в программировании.

Задание 1

Первое задание заключалось в создании функции, осуществляющей последовательный поиск (относительная сложность алгоритма O(n)), суть алгоритма заключается в пошаговом переборе всех элементов массива, от этого и такая высокая сложность. Реализация в коде приведена ниже (рис. 1).

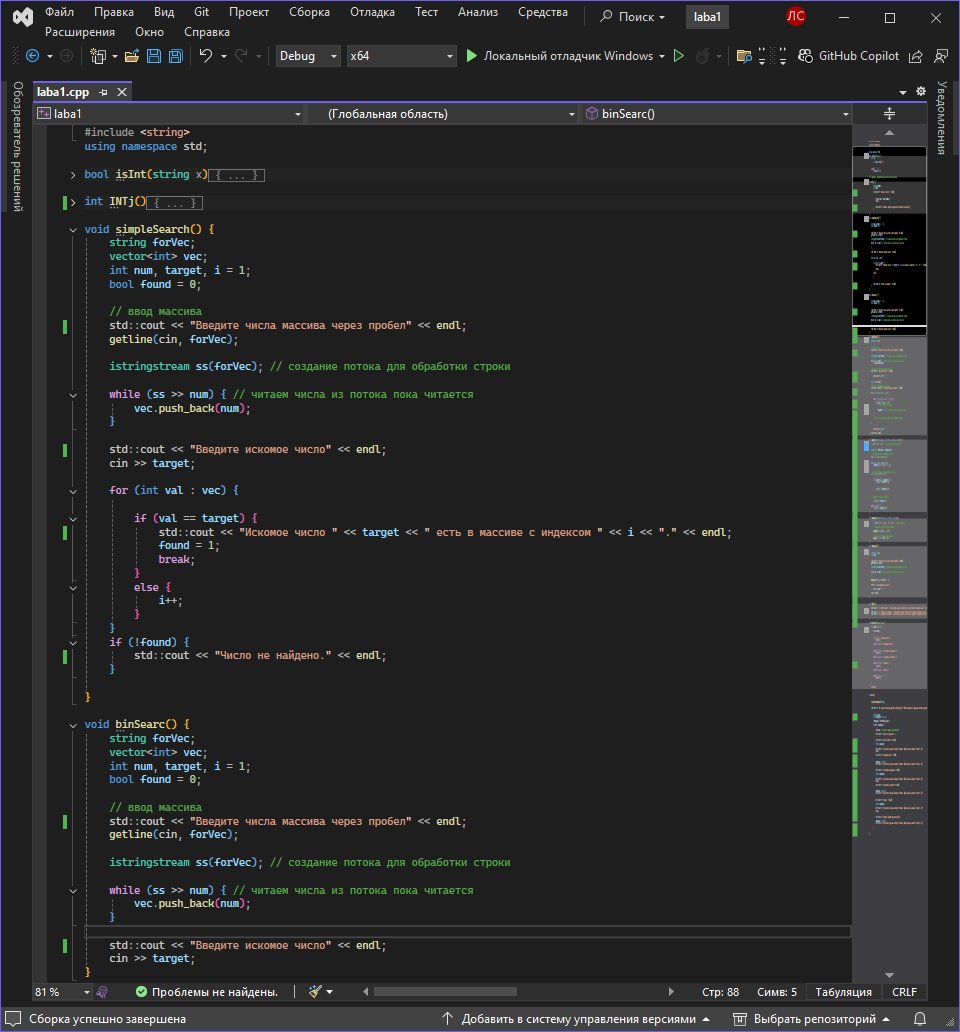


Рисунок 1

Задание 2

Второе задание заключалось в создании функции, осуществляющей бинарный поиск (относительная сложность алгоритма O(log(n))), суть алгоритма заключается в выборе среднего элемента отсортированного массива, сравнение искомого числа с средним и отсечением большей или меньшей половины (в зависимости от сравнения среднего числа с искомым), функция является рекурсивной. Реализация в коде приведена ниже (рис. 2).

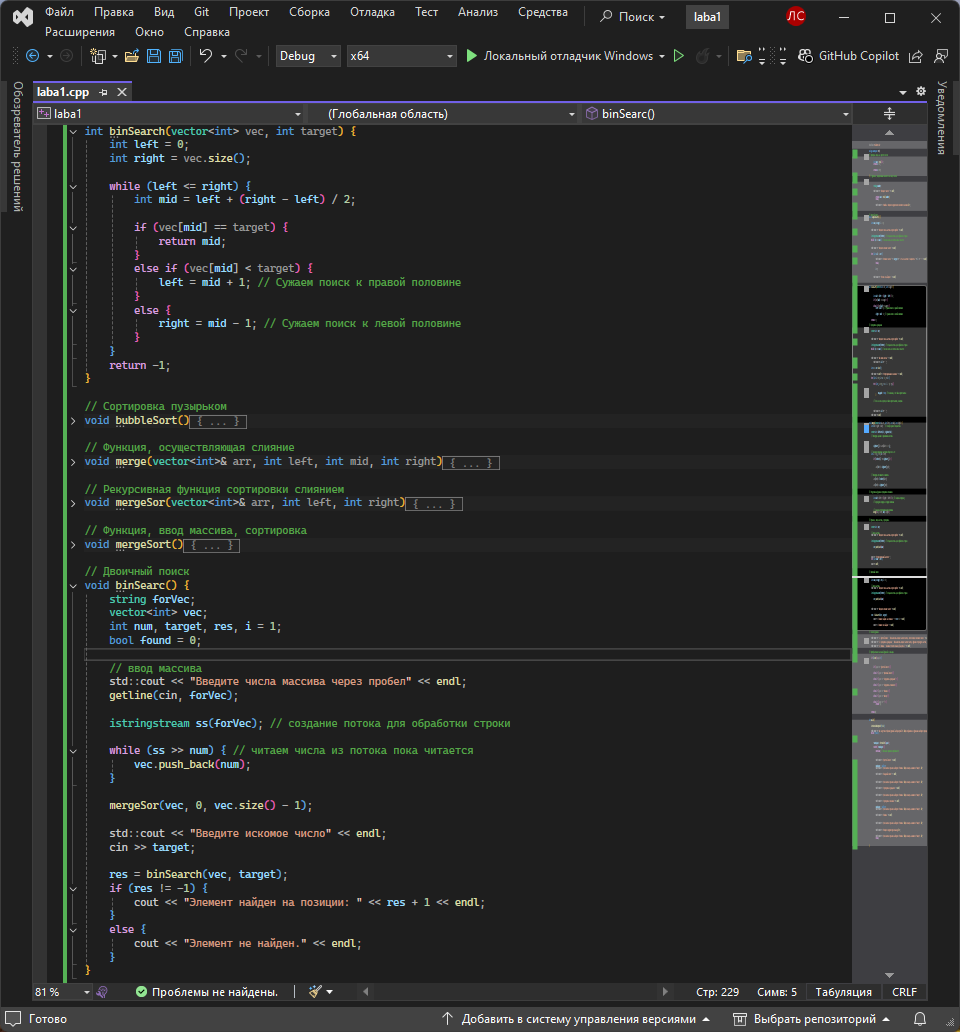


Рисунок 2

Задание 3

Третье задание заключалось в создании функции, осуществляющей сортировку пузырьком (относительная сложность алгоритма O(n2)), суть алгоритма в последовательном переборе и сравнении элементов массива (если левый элемент больше правого, то левый элемент перемешается вправо, так происходит до тех пор, пока массив не будет отсортирован. Реализация в коде приведена ниже (рис. 3).

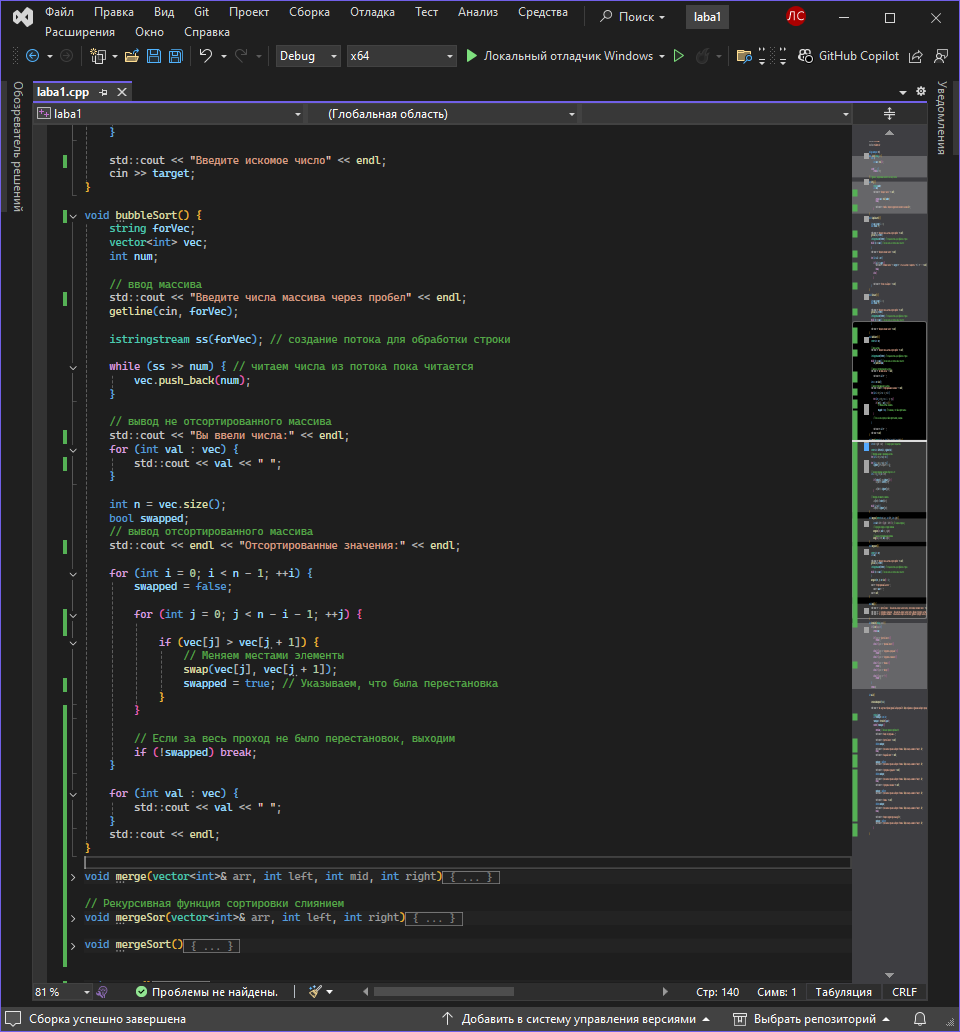


Рисунок 3

Задание 4

Четвертое задание заключалось в создании функции, осуществляющей сортировку слиянием (относительная сложность алгоритма O(n\*log(n))), суть алгоритма в перераспределении элементов массива следующим образом:

1. Если a = b, программа ничего не делает, поскольку подмассив уже отсортирован
2. Вычисляется позицию среднего элемента: k = [(a + b)/2].
3. Рекурсивно отсортировать подмассив array[a…k].
4. Рекурсивно отсортировать подмассив array[k + 1…b].
5. Слить отсортированные подмассивы array[a…k] и array[k + 1…b] в отсортированный подмассив array[a…b].

Реализация в коде приведена ниже (рис. 4-5).

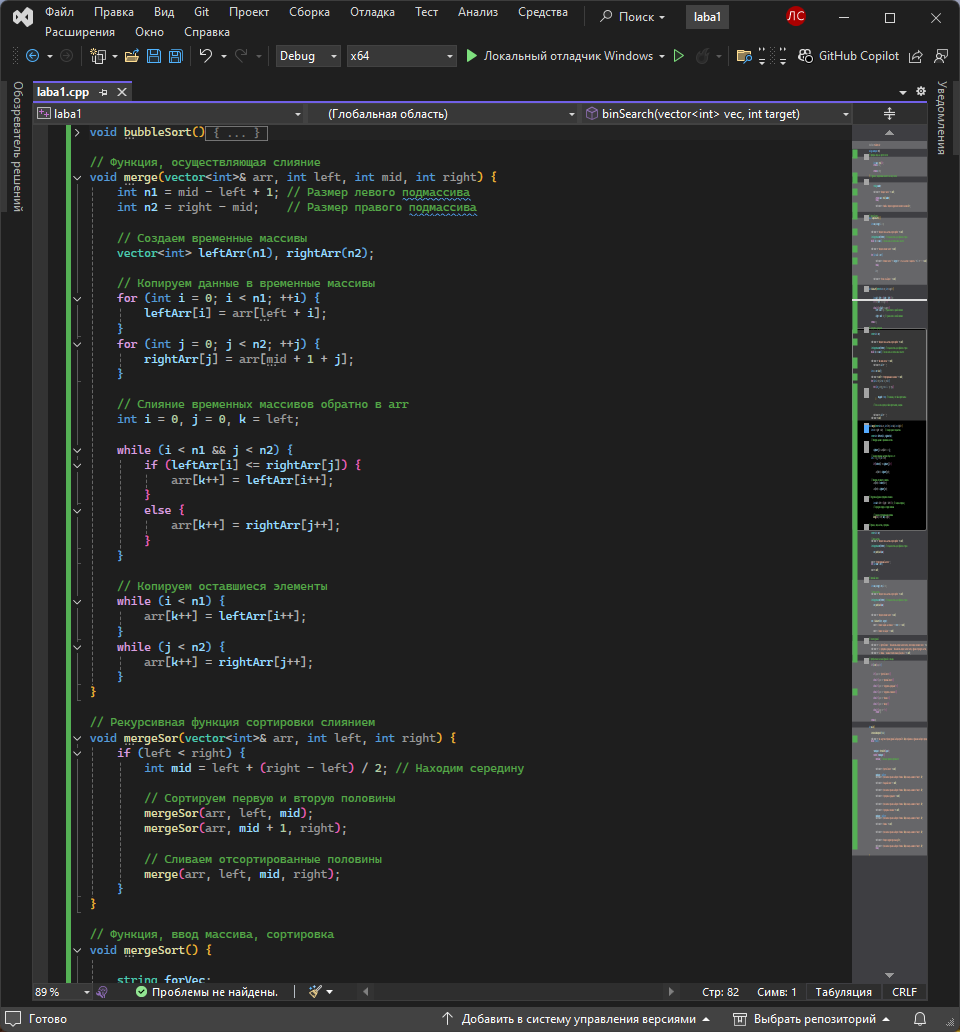


Рисунок 4

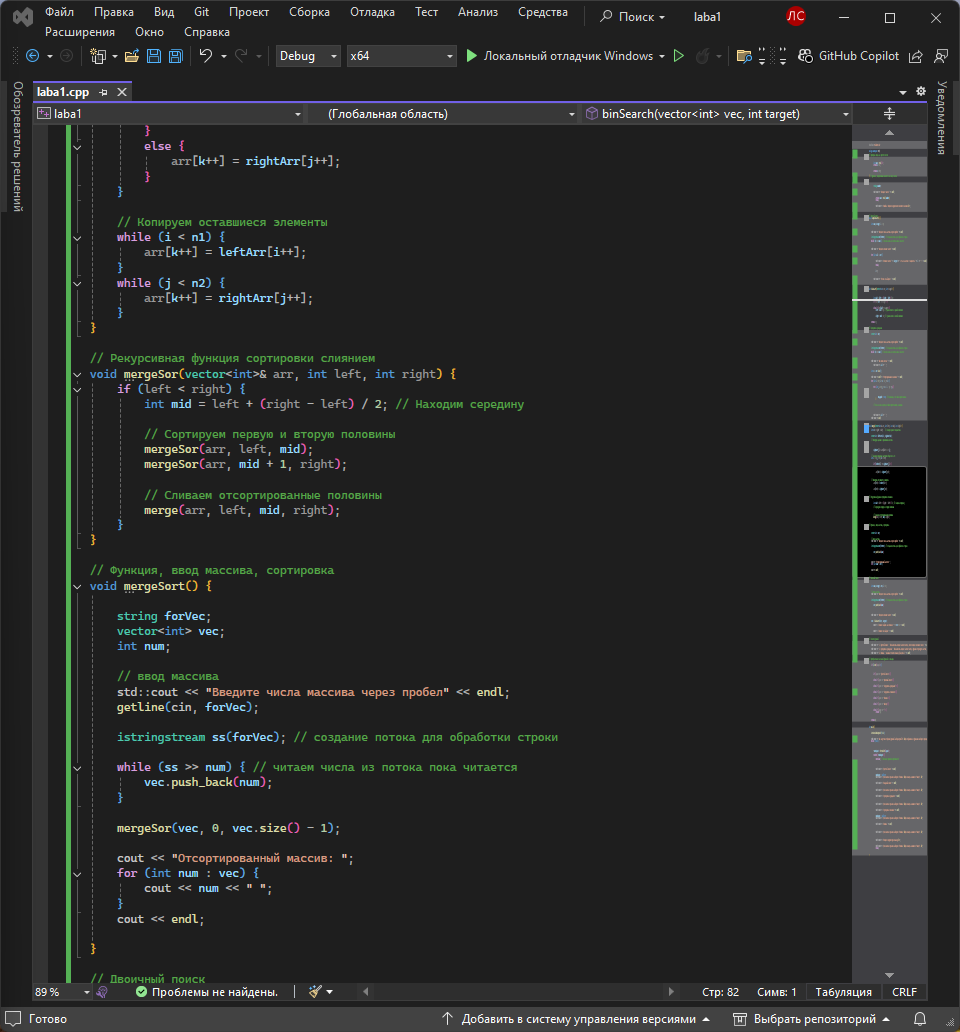


Рисунок 5

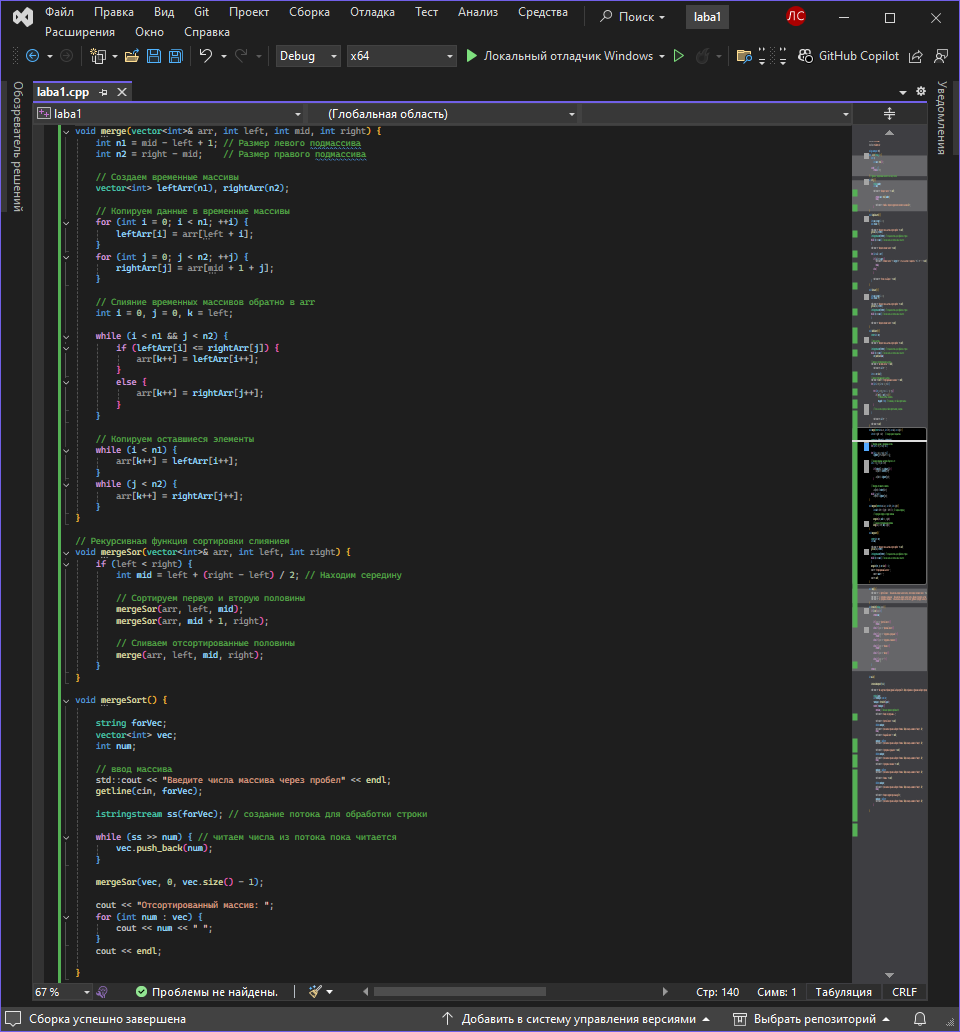


Рисунок 6

Вывод:

Ознакомился с понятием алгоритма, изучил различные виды алгоритмов и их применение в программировании.