Министерство образования и науки Российской Федерации

Федеральное государственное образовательное автономное учреждение высшего образования

"Пермский национальный исследовательский политехнический университет"

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №13

Дисциплина: Основы алгоритмизации и программирования

Тема: Стандартные обобщенные алгоритмы библиотеки STL.

Вариант 15

Пермь – 2023

|  |
| --- |
| Выполнил работу |
| Студент группы РИС-22-1б |
| Кирпичников Илья |
|  |
| Проверил работу |
| Доцент кафедры ИТАС |

**Постановка задачи**

Задача 1.

1. Создать последовательный контейнер.
2. Заполнить его элементами пользовательского типа (тип указан в варианте). Для пользовательского типа перегрузить необходимые операции.
3. Заменить элементы в соответствии с заданием (использовать алгоритмы
4. replace\_if(), replace\_copy(), replace\_copy\_if(), fill()).
5. Удалить элементы в соответствии с заданием (использовать алгоритмы
6. remove(),remove\_if(), remove\_copy\_if(),remove\_copy())
7. Отсортировать контейнер по убыванию и по возрастанию ключевого поля (использовать алгоритм sort()).
8. Найти в контейнере заданный элемент (использовать алгоритмы find(), find\_if(), count(), count\_if()).
9. Выполнить задание варианта для полученного контейнера (использовать алгоритм
10. for\_each()) .
11. Для выполнения всех заданий использовать стандартные алгоритмы библиотеки STL.

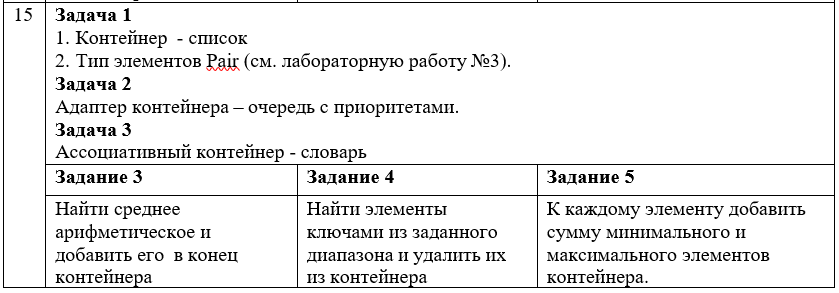
Задача 2.

1. Создать адаптер контейнера.
2. Заполнить его элементами пользовательского типа (тип указан в варианте). Для пользовательского типа перегрузить необходимые операции.
3. Заменить элементы в соответствии с заданием (использовать алгоритмы
4. replace\_if(), replace\_copy(), replace\_copy\_if(), fill()).
5. Удалить элементы в соответствии с заданием (использовать алгоритмы
6. remove(),remove\_if(), remove\_copy\_if(),remove\_copy())
7. Отсортировать контейнер по убыванию и по возрастанию ключевого поля (использовать алгоритм sort()).
8. Найти в контейнере элемент с заданным ключевым полем (использовать
9. алгоритмы find(), find\_if(), count(), count\_if()).
10. Выполнить задание варианта для полученного контейнера (использовать алгоритм
11. for\_each()) .
12. Для выполнения всех заданий использовать стандартные алгоритмы библиотеки STL.

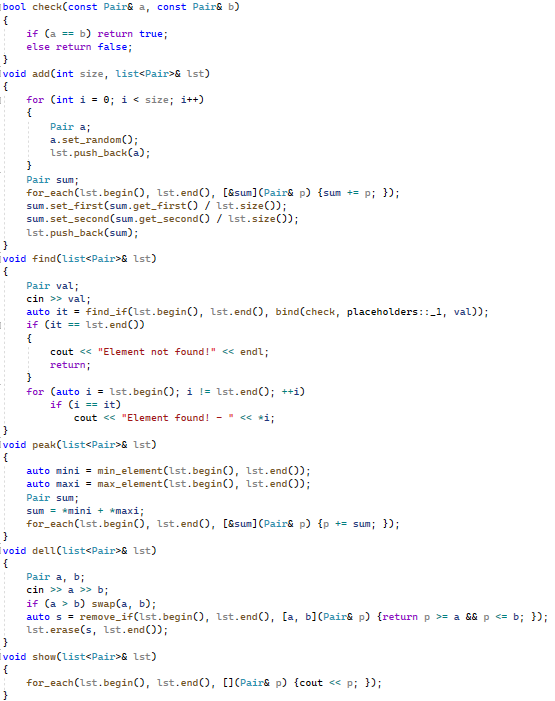
Задача 3

1. Создать ассоциативный контейнер.
2. Заполнить его элементами пользовательского типа (тип указан в варианте). Для пользовательского типа перегрузить необходимые операции.
3. Заменить элементы в соответствии с заданием (использовать алгоритмы
4. replace\_if(), replace\_copy(), replace\_copy\_if(), fill()).
5. Удалить элементы в соответствии с заданием (использовать алгоритмы
6. remove(),remove\_if(), remove\_copy\_if(),remove\_copy())
7. Отсортировать контейнер по убыванию и по возрастанию ключевого поля (использовать алгоритм sort()).
8. Найти в контейнере элемент с заданным ключевым полем (использовать
9. алгоритмы find(), find\_if(), count(), count\_if()).
10. Выполнить задание варианта для полученного контейнера (использовать алгоритм
11. for\_each()) .
12. Для выполнения всех заданий использовать стандартные алгоритмы библиотеки STL.

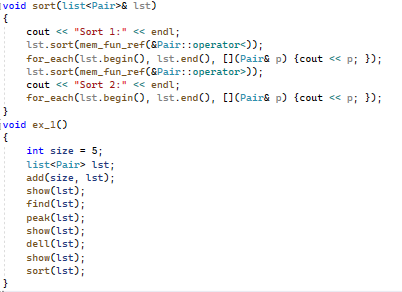
Вариант 15:



# Задание 1

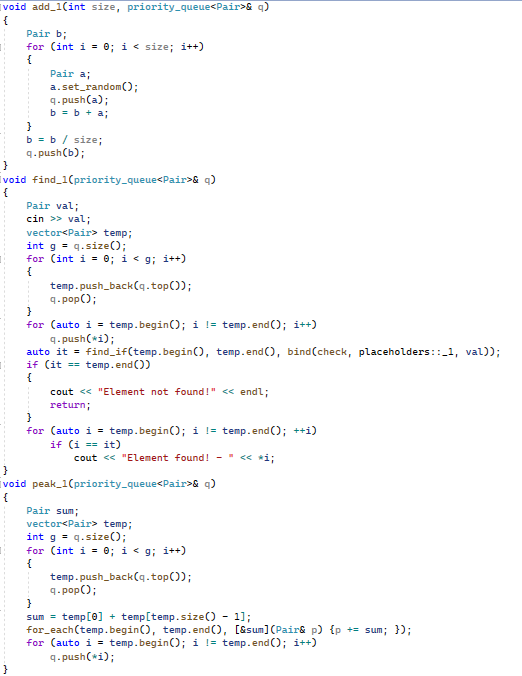
****

*Приложение 1 – Задание 1*

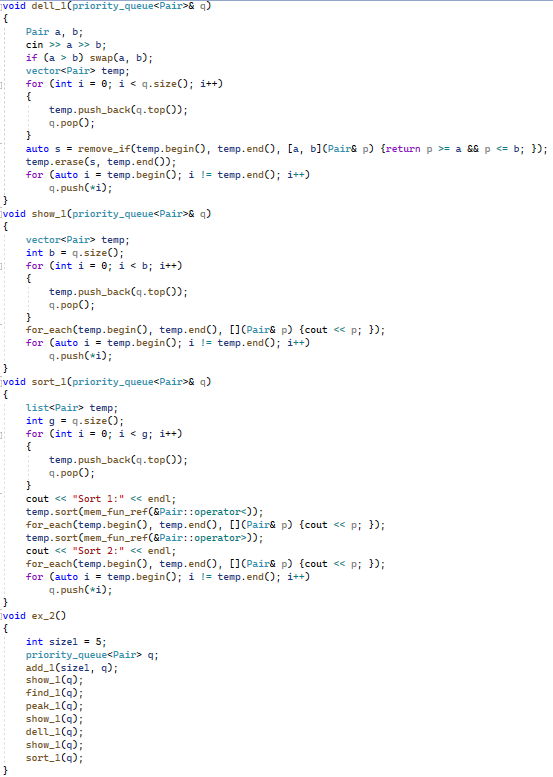
**

*Приложение 2 – Задание 1*

# Задание 2

****

*Приложение 3 – Задание 2*

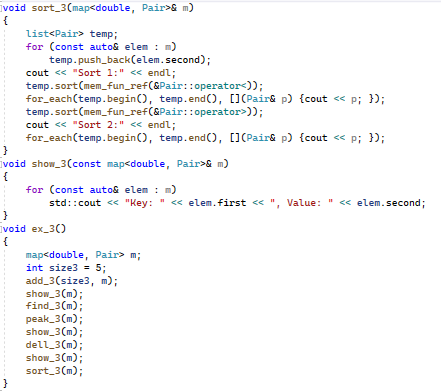
**

*Приложение 4 – Задание 2*

# Задание 3

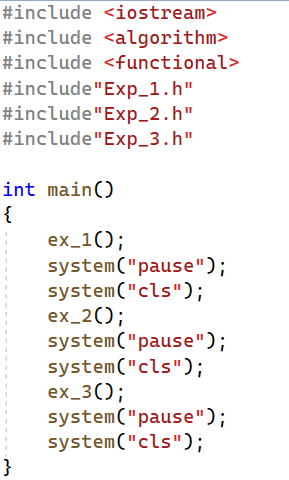
# 

*Приложение 5 – Задание 3*

**

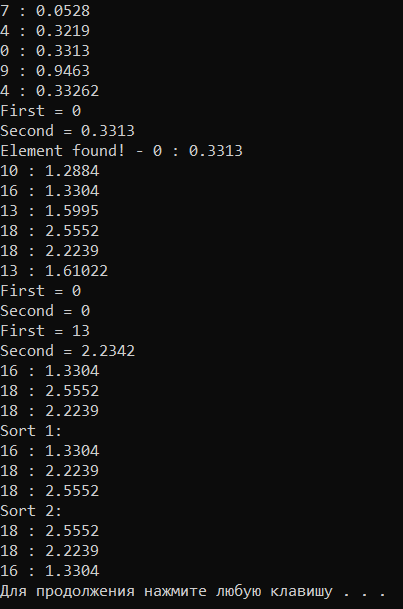
*Приложение 6 – Задание 3*

**Функция main()**

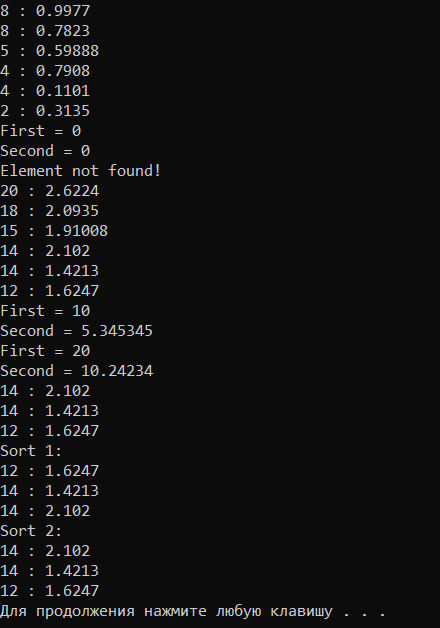
****

*Приложение 7 – Главная функция*

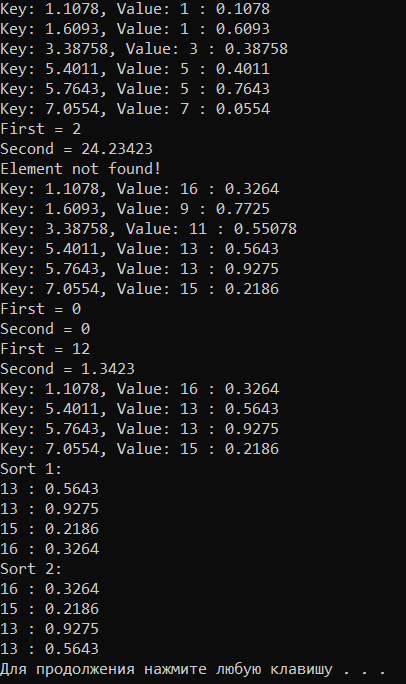
**Работа программы**



*Приложение 8 – Работа программы - задание 1*

**

*Приложение 9 – Работа программы – задание 2*

**

*Приложение 10 – Работа программы – задание 3*