**Лабораторная работа**

по дисциплине «Технологии программирования»

|  |  |
| --- | --- |
| Студент гр. 3530904/20003  Андреев В. И. |  |
| Руководитель  Ст. преподаватель | Череповский Д. К. |

**Оглавление**

[**Лабораторная работа №3: Бинарное дерево поиска** 3](#_Toc130578870)

[**Выполнение работы** 3](#_Toc130578871)

[**Задание 1. Реализуйте структуру данных «ограниченный» стек** 3](#_Toc130578872)

[**1.** **Общая постановка задачи** 3](#_Toc130578873)

[**2.** **Детальные требования, тест-план** 4](#_Toc130578874)

[**Приложение 1** 5](#_Toc130578875)

[**Список иллюстраций** 5](#_Toc130578876)

# **Лабораторная работа №1**

Цель: формирование навыков использования потоков, итераторов для решения поставленных задач.

**Выполнение работы.**

# **Задание 1. Потоки, итераторы, алгоритмы**

## **Общая постановка задачи**

Написать программу, которая выполняет следующие действия:

1. Заполняет std::vector структурами DataStruct, прочитанными со стандартного ввода. Чтение необходимо осуществлять с помощью итераторов и алгоритмов STL (std::copy, итераторов потока и перегрузки оператора побитового сдвига для чтения из потока)
2. Сортирует считанные данные следующим образом:

(a) По возрастанию key1

(b) По возрастанию key2, если key1 одинаковые

(c) По возрастанию длины строки key3, если прочие поля равны

1. Выводит результаты сортировки на стандартный вывод. Каждая строка должна содержать ровно один объект. Формат каждого выводимого объекта аналогичен формату ввода. Вывод необходимо осуществлять с помощью итераторов и алгоритмов STL (std::copy, итераторов потока и перегрузки оператора побитового сдвига для вывода в поток) Требования Входные данные могут содержать строки с неподдерживаемым форматом данных. Такие строки должны игнорироваться. Формат данных, который необходимо обрабатывать зависит в том числе от определения DataStruct
2. Тип DataStruct определён следующим образом:

struct DataStruct {

unsigned long long key1;

unsigned long long key2;

std::string key3;

};

Каждая запись ограничена парой скобок. Внутри этих скобок в качестве разделителей используются пробельные символы и символы :

Например: (:key1 10ull:key2 ’c’:key3 "Data":) Порядок описания полей в структуре не определён. Например, следующие структуры данных считаются идентичными: (:key1 10ull:key2 ’c’:key3 "Data":)

(:key2 ’c’:key1 10ull:key3 "Data":)

(:key3 "Data":key2 ’c’:key1 10ull:)

Имя поля и соответствующее значение гарантировано разделены ровно одним пробелом. Символы двоеточия гарантировано примыкают к прочим элементам записи. При выводе в поток поля выводятся в том порядке, в котором они заданы в структуре (запоминать исходный порядок не требуется)

1. Поля могут иметь следующий тип и соответствующий формат:

[ULL LIT] Беззнаковое максимально доступной ёмкости (unsigned long long) в формате литерала: :keyX 89ull: :keyX 89ULL:

[ULL BIN] Беззнаковое максимально доступной ёмкости (unsigned long long) в формате двоичного литерала: :keyX 0b1000101: :keyX 0B001001:

Вариант 1: key 1 - ULL LIT key 2 - ULL BIN

## **Приложение 1**

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2

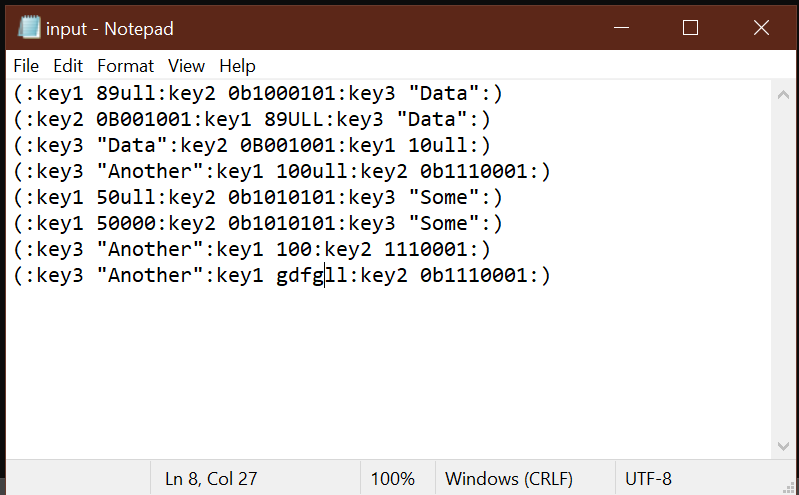


Рисунок 3

## **Список иллюстраций**

[Рисунок 1 5](#_Toc130569321)

[Рисунок 2 5](#_Toc130569322)

[Рисунок 3 5](#_Toc130569323)