

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

«Национальный исследовательский

Нижегородский государственный университет им. Н.И. Лобачевского»  
(ННГУ)

Институт информационных технологий, математики и механики

Кафедра математического обеспечения и суперкомпьютерных  
технологий

## **ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ**

**«Реализация класса упорядоченная таблица»**

**Выполнил:** студент группы  
381706-2

Крюков Дмитрий Алексеевич

\_\_\_\_\_ Подпись

**Научный руководитель:**

ассистент каф. МОСТ ИИТММ

Лебедев Илья Геннадьевич

\_\_\_\_\_ Подпись

Нижний Новгород

2019

## Содержание

<u>Содержание.....</u>	<u>2</u>
<u>1. Введение.....</u>	<u>3</u>
<u>2. Постановка задачи.....</u>	<u>4</u>
<u>3. Руководство пользователя.....</u>	<u>5</u>
<u>4. Руководство программиста.....</u>	<u>6</u>
<u>4.1 Описание структуры программы.....</u>	<u>6</u>
<u>4.2 Описание структур данных.....</u>	<u>6</u>
<u>4.3 Описание алгоритмов.....</u>	<u>7</u>
<u>5. Заключение.....</u>	<u>8</u>
<u>6. Литература.....</u>	<u>9</u>

# 1. Введение

Таблицы, в которых записи располагаются в порядке возрастания (или убывания) ключей, называются сортированными (упорядоченными)

Упорядоченность таблиц может быть организована только при возможности сравнения ключей (на множестве ключей задано отношение линейного порядка)

Операции под таблицей

- Поиск записи по ключу
- Вставка новой записи
- Удаление записи

Идея похода (пример) – упорядочение данных при большом объеме хранимой информации (например, алфавитный порядок в словарях)

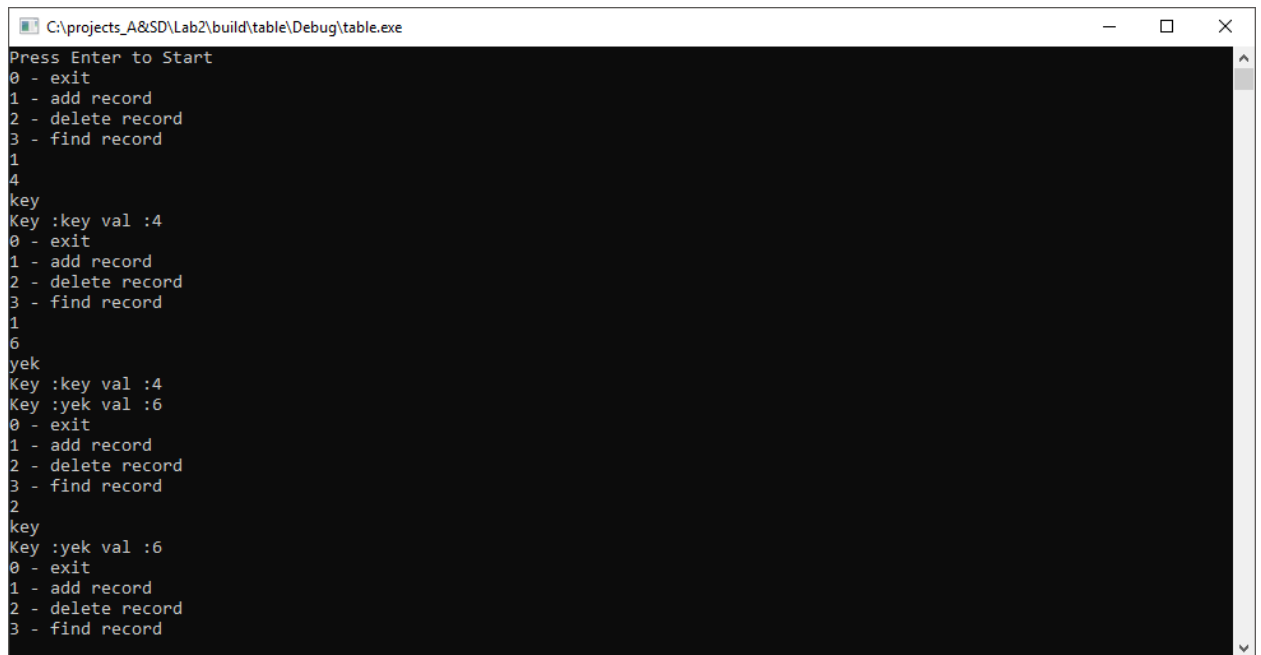
- В любом месте упорядоченных данных известно, в какой части набора располагаются искомые значения
- Отсутствие нужных данных может быть определено без полного просмотра

## 2. Постановка задачи

- Организация доступа по имени для управления информацией в привычной для человека форме
- упорядочение данных при большом объеме хранимой информации (например, алфавитный порядок в словарях)
- Представление данных во многих задачах из разных областей приложений (таблицы идентификаторов, номенклатура изделий, словари и т.п.)
- Абстрагирование от проблем распределения памяти при размещении данных
- Реализация концепции *ассоциативной памяти* (память с доступом к данным по содержимому в этих данных)
- Отображение на ЭВМ такого важного математического понятия как *множества*

### 3. Руководство пользователя

Данная программа предназначена для тестирования динамической структуры  
сортированная таблица



```
C:\projects_A&SD\Lab2\build\table\Debug\table.exe
Press Enter to Start
0 - exit
1 - add record
2 - delete record
3 - find record
1
key
Key :key val :4
0 - exit
1 - add record
2 - delete record
3 - find record
1
6
yey
Key :key val :4
Key :yey val :6
0 - exit
1 - add record
2 - delete record
3 - find record
2
key
Key :yey val :6
0 - exit
1 - add record
2 - delete record
3 - find record
```

пользователю предлагается набор команд :

- вставка записи в таблицу
- удаление записи из таблицы
- поиск записи в таблице

## 4. Руководство программиста

### 4.1 Описание структуры программы

1. Модуль sorttablelib (TSortTable.h) – реализация класса таблица
2. tablelib (TabRecord.h, TKey.h, TKey.cpp) – реализация класса запись таблицы, ключ
3. Модуль sorttable(main.cpp) – реализация программы для тестирования динамической структуры сортированная таблица
4. Модуль sortstabletest(sorttable\_test.cpp) - тестирование класса сортированная таблица при помощи Google C++ Testing Framework.

### 4.2 Описание структур данных

#### Структура

Поля:

dataCount — число записей в таблице

tabSize — максимальное число записей в таблице

pRecs – массив записей таблицы

SortType type — вид сортировки используемый таблицей

Методы:

Add(TTabRecord<ValType> \*tr) — добавляет запись в таблицу

Delete(TKey k) — удаляет запись из таблицы

Search(TKey k) — поиск по ключу

operator = (TScanTable &st) — преобразование просмотровой таблицы в сортированную

operator[] (TKey k) — обращение по ключу, в случае отсутствия записи добавляет пустую запись с заданным ключом

### 4.3 Описание алгоритмов

Алгоритм быстрой сортировки

Разделение массива с использованием ведущего элемента

```
key = K[0]
пока ( i1 <= i2 ) {
    пока ( (i1 < N) && (K[i1] <= key) ) i1++;
    пока ( K[i2] > key ) i2--;
    если( i1 < i2 )
        kt = K[i1]
        K[i1] = K[i2]
        K[i2] = kt; }
K[0] = K[i2]
K[i2] = key;
i = i2
```

## **5. Заключение**

В ходе работы был реализован класс сортированная таблица, в нем реализованы функции удаления, вставки и поиска, а так же добавлена индексация по ключу



## **6. Литература**

1. Гергель В.П. Методические материалы по курсу «Методы программирования 2», Нижний Новгород, 2015.