



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**Отчет по лабораторной работе №2**  
**«ЗАПИСИ С ВАРИАНТАМИ. ОБРАБОТКА ТАБЛИЦ»**

Студент Дьяченко Артём Александрович

Группа ИУ7 – 33Б

Преподаватель Барышникова М. Ю.

# Оглавление

<u>ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ.....</u>	<u>3</u>
<u>ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ.....</u>	<u>3</u>
<u>НАБОР ТЕСТОВ.....</u>	<u>5</u>
<u>ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ.....</u>	<u>6</u>
<u>ВРЕМЕННЫЕ ЗАМЕРЫ.....</u>	<u>8</u>
<u>ОБЪЁМ ЗАНИМАЕМОЙ ПАМЯТИ.....</u>	<u>8</u>
<u>ОЦЕНКА ЭФФЕКТИВНОСТИ.....</u>	<u>8</u>
<u>ОПИСАНИЕ АЛГОРИТМА.....</u>	<u>9</u>
<u>ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ.....</u>	<u>10</u>
<u>ВЫВОД.....</u>	<u>10</u>

## ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ

Создать таблицу, содержащую не менее 40-ка записей (тип – запись с вариантами (объединениями)). Упорядочить данные в ней по возрастанию ключей, двумя алгоритмами сортировки, где ключ – любое невариантное поле (по выбору программиста), используя: а) саму таблицу, б) массив ключей. (Возможность добавления и удаления записей в ручном режиме обязательна). Осуществить поиск информации по варианту.

Ввести список стран, в которых можно отдохнуть, содержащий название страны, количество жителей, столицу, материк, нужна ли прививка или ПЦР, основной вид туризма (экскурсионный - количество объектов, основной вид (природа, история, искусство); пляжный – основной сезон, температура воздуха и воды, время полета до страны; спортивный – вид спорта (горные лыжи, серфинг, восхождения), минимальная стоимость отдыха,). Вывести список стран на выбранном материке, где можно заняться указанным видом спорта.

## ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ

### **Входные данные:**

**Файл с данными:** первая строка содержит названия полей.

Остальные: значения полей, разделённые пробелом. Если информация в поле отсутствует, оно помечается символом « – ».

**Целое число:** задаёт пункт меню.

**Строковые и числовые данные:** для добавления/удаления параметров записей.

### **Выходные данные:**

Текущая таблица, исходная таблица, отсортированная исходная таблица, отсортированная таблица ключей, сравнение эффективности сортировок, результат поиска по таблице,

### **Функции меню программы:**

1. Выгрузка данных из файла.
2. Просмотр текущей таблицы.
3. Добавление информации о новой стране в конец таблицы.
4. Удаление информации о стране с заданным названием.
5. Просмотр отсортированной по названию страны таблицы ключей.
6. Просмотр упорядоченной по названию страны таблицы.
7. Просмотр исходной таблицы в упорядоченном по названию страны виде по упорядоченной таблице ключей.
8. Вывод списка стран на выбранном материке, где можно заняться указанным видом спорта.

### **Обращение к программе:**

Запускается через терминал: `./app.exe название_файла.`

### **Аварийные ситуации:**

1. Неверные аргументы командной строки.  
«Ошибка: Неверные аргументы командной строки»
2. Ввод некорректного пункта меню.  
«Ошибка: Введён некорректный пункт меню»
3. Запрос на удаление несуществующей записи.  
«Ошибка: Страны с таким названием в списке нет!»
4. Ввод неверных данных страны для ввода и/или выходящие за границы значений.
5. Ввод в числовые данные другой тип.  
«Ошибка: Требуется целое число!»

## НАБОР ТЕСТОВ

№	Название теста	Пользовательский ввод	Вывод
1	Некорректный ввод пункта меню	abc	«Ошибка: Введён некорректный пункт меню»
2	Некорректный ввод пункта меню	112	«Ошибка: Введён некорректный пункт меню»
3	Просмотр текущей таблицы	2	Текущая таблица
4	Неверные аргументы командной строки	./app.exe wrong_file	«Ошибка: Не удалось открыть файл!»
5	Корректный файл	1	«Данные успешно загружены»
6	Корректное добавление страны	China 120000000 Seul Asia 1 Excursion History	«Данные о стране успешно добавлены»
7	Некорректное добавление страны	China 120000000 Seul Asia 1 Excursion History 12 1234	«Ошибка: Требуется целое число»
8	Запрос на удаление несуществующей страны.	4 Uganda	«Ошибка: Страны с таким названием в списке нет!»
9	Отсутствие стран, удовлетворяющим условиям поиска	8 Asia Climb	«Ошибка: Подходящей страны не найдено!»

## ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ

```
typedef struct
{
    char name[MAX_NAME_LEN];
    int index;
} country_key_t;
```

name – название страны

index – индекс ячейки, в

которой содержится страна

```
typedef struct
{
    country_key_t key[MAX_TABLE_LEN];
    size_t len;
} country_key_table_t;
```

key – массив ключей

len – длина массива ключей

Структура *таблицы ключей* и самих *ключей*.

```
typedef union
{
    excursion_t excursion;
    beach_t beach;
    sport_t sport;
} info_t;
```

```
typedef struct
{
    char name[MAX_NAME_LEN];
    int people;
    char capital[MAX_CAPITAL_LEN];
    char mainland[MAX_MAINLAND_LEN];
    int pcr;
    char type[MAX_TYPE_LEN];
    info_t info;
} country_t;
```

name – название страны

people – население страны

capital — столица

mainland — материк

pcr — нужен ли ПЦР-тест

type — тип отдыха в стране

info — объединение структур,

информации об отдыхе в

данной стране

```
typedef struct
{
    country_t country[MAX_TABLE_LEN];
    size_t len;
} country_table_t;
```

country — массив стран

len — кол-во стран в таблице

Структура *таблицы стран* и самих *стран*.

Объединение **info\_t** содержит в себе три структуры под все типы отдыха: *Excursion*, *Beach*, *Sport*.

```
typedef struct
{
    int obj_count;           obj_count - кол-во объектов для посещения
    char type[MAX_TYPE_LEN]; type — тип объектов
} excursion_t;

typedef struct
{
    char season[SEASON_LEN]; season - основное время года
    int temp_air;             temp_air - температура воздуха
    int temp_water;          temp_water - температура воды
    int arriving_time;       arriving_time - время полёта
} beach_t;

typedef struct
{
    char type[MAX_TYPE_LEN]; type - тип спорта
    int cost;                type — стоимость
} sport_t;
```

## ВРЕМЕННЫЕ ЗАМЕРЫ (МС)

Кол-во записей	Сортировка пузырьком		Быстрая сортировка	
	Основная таблица	Таблица ключей	Основная таблица	Таблица ключей
40	33	31	12	11
200	901	120	37	16
500	4903	741	181	24
1000	14187	8500	643	46
5000	173597	73445	1965	278

## ОБЪЁМ ЗАНИМАЕМОЙ ПАМЯТИ (БАЙТ)

Кол-во записей	Основная таблица	С таблицей ключей
40	2648	3216
200	13208	16016
500	33008	40016
1000	66008	80016
5000	330008	400016



## ОЦЕНКА ЭФФЕКТИВНОСТИ

Кол-во записей	Отношение времени сортировки <b>пузырьком</b> основной таблицы к таблице ключей	Отношение времени <b>быстрой</b> сортировки основной таблицы к таблице ключей
40	1.06	0.91
200	7.5	1.06
500	6.62	7.54
1000	1.67	13.98
5000	2.36	7.07

Во временных данных приведены средние значения за 100 замеров.

Столь непонятный паттерн, разные отношения (в методе пузырька) обусловлены тем, что тестовые данные генерировались случайным образом. А аномалия в первом отношении быстрой сортировки обусловлена маленьким набором данных.

Процент объёма занимаемой памяти таблицы ключей не превышает 18%, но при этом скорость работы алгоритма с ключами увеличивается на много больший процент: в среднем на 280%.

Сортировка пузырьком основной таблицы медленней быстрой сортировки в среднем в 33 раза.

## ОПИСАНИЕ АЛГОРИТМА

1. Программа запускается с одним ключом — названием файла со списком.
2. Пользователь вводит пункт меню.
3. Если пункт меню связан с записью/вводом данных — проводится проверка данных и выполняется запрос.

4. Если пункт меню связан с выводом отсортированной таблицы – список стран/ключей сортируется методом быстрой сортировки и выводится на экран.
5. В случае возникновения аварийной ситуации программа оповещает пользователя о возникшей ошибке и завершается.

## **ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ**

### **1. Как выделяется память под вариантную часть записи?**

Выделяется такое количество байт, чтобы хватило на самую большую структуру (страна с типом отдыха Beach в нашем случае)

### **2. Что будет, если в вариантную часть ввести данные, несоответствующие описанным?**

Программа завершит работу и выдаст сообщение об ошибке.

### **3. Кто должен следить за правильностью выполнения операций с вариантной частью записи?**

Программист должен реализовать программу так, чтобы пользователь, введя неверные данные, получил сообщение об ошибке и невыполненную операцию. В свою очередь, пользователь, если он хочет корректного результата, должен следить за корректностью вводимых данных.

### **4. Что представляет собой таблица ключей, зачем она нужна?**

Таблица ключей — список ключей и длина этого списка, где каждый ключ содержит название страны и индекс, где в основной таблице эта страна расположена.

### **5. В каких случаях эффективнее обрабатывать данные в самой таблице, а когда – использовать таблицу ключей?**

Во всех случаях эффективнее использовать таблицу ключей. Но при малых значениях разница будет незаметна.

### **6. Какие способы сортировки предпочтительнее для обработки таблиц и почему?**

Те, что подразумевают использование компаратора: с ним удобно работать, т.к. для сравнения каждой структуры достаточно написать функцию сравнения. А вообще, чем способ сортировки быстрее – тем лучше.

## Вывод

В результате выполнения этой лабораторной работы, которая стала самой объёмной за всё время моего обучения (~1000 строк кода), я улучшил свои навыки в декомпозиции, структуризации, работе со структурами, объединениями, указателями и внешними библиотеками.

В результате оценки эффективности, я пришёл к выводу, что ультимативным решением будет **всегда** использовать сортировки с меньшей сложностью, т.к. при большом кол-ве данных они дадут значительный выигрыш во времени. Аналогично всегда стоит использовать список ключей, чтобы при сортировке менять лишь индексы стран, а не перемещать все данные из одной ячейки памяти в другую.