|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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 Егорова Полина Александровна

Группа ИУ7 – 34Б

Преподаватель Силантьева Александра Васильевна

2021 г.

# **Описание условия задачи**

Создать таблицу, содержащую не менее 40-ка записей (тип – запись с вариантами). Упорядочить данные в ней по возрастанию ключей, где ключ – любое невариантное поле (по выбору программиста), используя:

* саму таблицу
* массив ключей

(возможность добавления и удаления записей в ручном режиме обязательна).

Ввести репертуар театров, содержащий: название театра, спектакль, режиссер, диапазон цены билета, тип спектакля: детский – для какого возраста, тип (сказка, пьеса); взрослый – пьеса, драма, комедия); музыкальный – композитор, страна, минимальный возраст, продолжительность). Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной.

# **Описание технического задания**

Входные данные:

1. Файл с данными: текстовый файл формата TXT. Разделителем в файле является символ пробел “ ”. Каждая новая запись таблицы обязательно должна находиться на новой строке.
2. Целое число - номер команды: целое число в диапазоне от 1 до 11.
3. Дополнения к таблице: строковое или целочисленное поле (в зависимости от команды)

Выходные данные:

1. Полученная таблица в отсортированном или неотсортированном виде (в зависимости от выполненной команды).
2. Таблица из отобранных по заданию записей основной таблицы.
3. Характеристика сравнения вариантов сортировки таблицы.

Функции программы:

1. Загрузить таблицу из файла (theatre.txt)
2. Вывести таблицу на экран.
3. Вывести на экран таблицу из всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной.
4. Добавить запись в конец таблицы.
5. Удалить запись из таблицы по названию спектакля.
6. Отсортировать таблицу (выбор сортировки и сортируемого параметра) и вывести ее на экран.
7. Отсортировать таблицу ключей (быстрая сортировка) и вывести ее на экран.
8. Сравнить время быстрой сортировки самой таблицы и таблицы ключей и сортировки пузырьком самой таблицы и таблицы ключей.
9. Очистить таблицу.
10. Завершить программу.
11. Отобразить таблицу значений ключей, их индексов в таблице ключей и в основной таблице.

Обращение к программе:

Запуск через терминал (./app.exe).

Аварийные ситуации:

1. Некорректный ввод номера команды: пустой ввод или число, большее чем 10, или меньшее, чем 1.

Сообщение на выходе: *«Incorrect number of command.»*

1. Некорректный файл.

Сообщение на выходе: *«File errors.»*

1. Работа с незаполненной таблицей.

Сообщение на выходе: *«Table is empty..»*

1. Некорректный ввод ответа “y/n”: строка, содержащая больше одного символа.

Сообщение на выходе: *«Incorrect input.»*

1. Некорректный ввод ответа “y/n”: строка, содержащая символы, отличные от “y” или “n”.

Сообщение на выходе: *«You have to enter ‘y’ or ‘n’ .»*

1. Некорректный ввод диапазона цен: минимальная цена больше максимальной.

Сообщение на выходе: *«Minimum price must be less than maximum.»*

# **Описание структуры данных**

Используется table\_r table - структура, включающая массивы структур playbill\_r playbill[COUNT\_RECORDS] и key\_r key[COUNT\_RECORDS] и размер таблицы size.

#define COUNT\_RECORDS 500  
#define MAX\_LEN\_NAME 50  
#define RANGE\_PRICE 2

**сhar theatre[MAX\_LEN\_NAME];** *//строка, в которую происходит считывание*

**char performance[MAX\_LEN\_NAME];** *//строка, в которую происходит считывание*

**char surname\_director[MAX\_LEN\_NAME];** *//строка, в которую происходит считывание*

**int min\_price, max\_price;** *//переменная, в которую происходит считывание*

**int child, adult, mus;** *//переменная, в которую происходит считывание*

// структура, содержащая подробную информацию о музыкальном спектакле  
typedef struct musical  
{  
 char composer[MAX\_LEN\_NAME];  
 char country[MAX\_LEN\_NAME];  
 int min\_age;  
 int duration;  
} musical\_r;

// структура, содержащая информацию о типе взрослого спектакля  
typedef struct adult  
{  
 int is\_comedy;  
 int is\_fairy\_tale;  
 int is\_play;  
} adult\_r;

// структура, содержащая информацию о типе детского спектакля  
typedef struct type\_childlike  
{  
 int is\_fairy\_tale;  
 int is\_play;  
} type\_childlike\_r;

// структура, содержащая подробную информацию о детском спектакле  
typedef struct childlike  
{  
 int age;  
 type\_childlike\_r type\_childlike;  
} childlike\_r;

// объединение, содержащее подробную информацию о виде спектакля  
typedef union type\_perfomance  
{  
 childlike\_r childlike;  
 adult\_r adult;  
 musical\_r musical;  
} type\_perfomance\_r;

// структура, содержащая сведения о типе спектакля (детский, взрослый, музыкалный)  
typedef struct choose\_type  
{  
 int is\_childlike; // детский  
 int is\_adult; // взрослый  
 int is\_musical; // музыкальный  
} type\_r;

// структура, содержащая все сведения о спектакле  
typedef struct playbill  
{  
 char name\_theatre[MAX\_LEN\_NAME]; // название театра  
 char name\_performance[MAX\_LEN\_NAME]; // название спектакля  
 char surname\_director[MAX\_LEN\_NAME]; // фамилия режиссера  
 int price\_range[RANGE\_PRICE]; // диапазон цен  
 type\_r choose\_type;  
 type\_perfomance\_r type\_perfomance;  
} playbill\_r;

// ключи  
typedef struct keys  
{  
 int key; // значение минимальной цены  
 int ind; // индекс записи с соответствующим значением цены в основной таблице  
} key\_r;

// таблица  
typedef struct table  
{  
 playbill\_r playbill[COUNT\_RECORDS];  
 key\_r key[COUNT\_RECORDS]; //минимальная цена  
 int size;  
} table\_r;

# **Описание алгоритма**

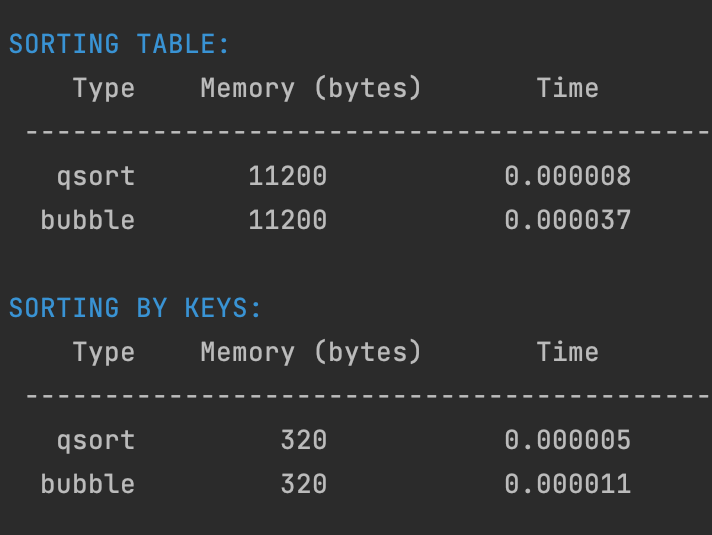
1. Ввод команды (возможные команды представлены в меню).
2. Пока пользователь не введет 10 (выход из программы), ему будет предложено вводить номера команд и выполнять действия по выбору.

# **Набор тестов**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Название теста** | **Пользовательский ввод** | **Результат** |
| 1 | Некорректный ввод команды | 45 | Incorrect number of command. |
| 2 | Несуществующий файл | 00.txt | File errors. |
| 3 | Добавление записи при максимальном размере таблицы |  | Table is empty. |
| 4 | Некорректный ввод численного поля (буква) | q | Incorrect input. |
| 5 | Длина строки превышает максимально возможную | Abcdefghijklmnopq | Incorrect input. |
| 6 | Минимальная цена больше мксимальной | 500  400 | Minimum price must be less than maximum. |
| 7 | Неверный ввод ответа формата “y/n” (что-либо кроме “y”, “n”) | Yes | You have to enter 'y' or 'n'. |
| 8 | Неверный ввод ответа формата “y/n” (что-либо кроме “y”, “n”) | No | You have to enter 'y' or 'n'. |
| 9 | Работа с пустой таблицей | (вызов любой команды до заполнения таблицы (2)) | Table is empty. |
| 10 | Ошибка при считывании данных из файла | (пропуск одного из параметров в записи поля, недопустимый символ/строка) | Error in writing in the file. |
| 11 | Некорректный ввод минимального возраста | -1 | You have to enter the non-negative integer number. |
| 12 | Некорректный ввод минимального возраста | Р | You have to enter the non-negative integer number. |
| 13 | Некорректный ввод минимального возраста (детский спектакль) | 19 | You have to enter the number less than 18. |
| 14 | Некорректно считанный минимальный возраст детского спектакля |  | Invalid age. |
| 15 | Некорректно считанный тип детского или взрослого спектакля |  | Invalid type. |
| 16 | Считан не один тип взрослого спектакля |  | Ambiguous choiсe. |
| 17 | Некорректно считан композитор или страна музыкального спектакля |  | Invalid composer/country. |
| 18 | Некорректно считан минимальный возраст или продолжительность музыкального спектакля |  | Invalid minimum age/duration |
| 19 | Некорректно считано название театра |  | Invalid theatre |
| 20 | Некорректно считано название спектакля |  | Invalid performance |
| 21 | Некорректно считана фамилия режиссера |  | Invalid director surname |
| 22 | Некорректно считана минимальная цена |  | Invalid min price |
| 23 | Некорректно считана максимальная цена |  | Invalid max price |
| 24 | Считанная минимальная цена больше сичтанной максимальной |  | Minimum price must be less than maximum. |
| 25 | Считан не один тип спектакля |  | Ambiguous choiсe. |
| 26 | Не выбран ни один из типов сортировки |  | You need choose only one type of sorting. |
| 27 | Выбраны несколько типов сортировки |  | You need choose only one type of sorting. |
| 28 | Не считан или некорректен размер таблицы |  | Incorrect data about the size of the table |
| 29 | Не найдены спектакли, с введенным названием (для удаления) |  | Fields not found. |
| 30 | Спектакли, с минимальным возрастом, равным введенному, не найдены |  | Fields not found. |
| 31 | Спектакли, с продолжительностью, меньше введенной, не найдены |  | Fields not found. |
| 32 | Таблица загружена успешно |  | Data is loaded into a table. |
| 33 | Введенные данные успешно добавлены в таблицу |  | Data is loaded into a table. |
| 34 | Поля успешно удалены из таблицы |  | Fields have been deleted. |
| 35 | Все данные успешно удалены из таблицы |  | Table is cleared. |

# **Сравнительная характеристика сортировок**

В лабораторной работе были рассмотрены и сравнены два вида сортировки – пузырьком и qsort. Замеры были проведены на списке структур, содержащем 40 записей. Время в секундах.



Вывод: массив ключей уменьшает время сортировки на 37.5% в случае сортировки пузырьком и на 70.2% в случае быстрой сортировки.

# **Ответы на контрольные вопросы**

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

Выделяется общий блок памяти для всех полей вариантной части. Размер памяти равен максимальному по длине полю вариантной части.

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

Невозможно найти подходящий тип из встречающих в записи вариантной части, поэтому считать данные невозможно. Поведение не определено.

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

За правильностью выполнения операций с вариантной частью должен следить программист.

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

Таблица ключей – массив или структура, содержащая индексы определенного элемента каждой записи исходной таблицы. Таблица ключей позволяется сократить время при сортировке и поиске записей в исходной таблице, так как хранит только один параметр.

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

Таблица ключей позволяет сэкономить время при сортировке таблицы, так как перестановка записей исходной таблицы не происходит. Данный подход имеет минусы: для размещения таблицы ключей требуется дополнительная память, а если ключом является символьное поле, его обработка требует дополнительных временных затрат (в цикле). Обработка данных происходит менее ресурсозатратно при работе с самой таблицей (а не таблицей ключей), если она содержит небольшое число записей.

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

Если будет производится сортировка самой таблицы, то необходимо использовать алгоритмы, требующие наименьшее количество операций перестановки. Если же сортировка производится по таблице ключей, то эффективнее использовать сортировки с наименьшей сложностью работы.

# **Вывод**

Существует большое количество способов обрабатывать информацию в таблицах, реализовывать сортировку, поиск. И выбор оптимального варианта ложится на плечи программиста, так как он должен уметь оценить объем обрабатываемых данных в таблице в целом, в одной ее записи, уметь подобрать подходящий параметр сортировки и тд. А вариантная часть записи помогает сократить расход памяти. Умение грамотно работать с вариантной частью и оптимально подбирать способ сортировки – важное умение программиста.