|  |  |
| --- | --- |
| **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 Ковалец Кирилл Эдуардович

Группа ИУ7 – 33Б

*2020 г.*

# **Описание условия задачи**

Создать таблицу, содержащую не менее 40-ка записей (тип – запись свариантами (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.

Мое задание по 8-ому варианту:

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:**

**Текстовый файл формата txt.** Каждая новая запись таблицы в обязательном порядке должна находиться на новой строке файла.

**Каждая запись должна содержать:**

1) Название театра;

2) Название спектакля;

3) Фамилия режиссёра;

4) Минимальная цена билета;

5) Максимальная цена билета;

6) Тип спектакля:

6.1. (child);

- минимальный возраст;

- тип детского спектакля (story, play);

6.2. (adult);

- тип спектакля для взрослых(play, drama, comedy);

6.3. (musical);

- фамилия композитора;

- название страны;

- минимальный возраст;

- продолжительность.

**Каждая запись должна удовлетворять следующим правилам:**

1. Максимальная длина названия театра - 23 символа;

2. Максимальная длина названия спектакля - 20 символов;

3. Максимальная длина фамилии режиссёра - 20 символов;

4. Минимальная цена билета > 0;

5. Максимальная цена билета > 0 и > Минимальной цены;

6. Возможно только три типа спектакля: child, adult, musical;

7. Минимальный возраст ребёнка - целое число в промежутке от 0 до 17 лет;

8. Возможно только два типа спектакля для детей: story, play;

9. Возможно только три типа спектакля для взрослых: play, drama, comedy;

10. Максимальная длина фамилии композитора - 20 символов;

11. Максимальная длина названия страны - 15 символа;

12. Минимальный возраст для посещения музыкального спектакля - целое число в промежутке от 0 до 23 лет;

13. Продолжительность музыкального спектакля должна быть больше нуля;

14. Допустимые символы в словах:

- латинские буквы от 'a' до 'z';

- латинские буквы от 'A' до 'Z';

- символы '-', '\_' не в начале слова ('\_' вместо символа ' ');

15. Слова в строке должны быть разделены пробелами.

**Выходные данные:**

Текущее состояние таблицы, результаты сравнения эффективности сортировок, результаты поиска по заданным полям.

**Команды:**

1. Загрузить список спектаклей из файла;

2. Вывести таблицу;

3. Добавить спектакль в конец таблицы;

4. Удалить спектакли, минимальная цена билетов которых больше указанного значения;

5. Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной;

6. Вывести массив ключей (Ключ - название спектакля);

7. Отсортировать массив ключей методом 'qsort' (Ключ - название спектакля);

8. Отсортировать массив ключей методом 'выбором' (Ключ - название спектакля);

9. Отсортировать таблицу методом 'qsort' (по названию спектакля);

10. Отсортировать таблицу методом 'выбором' (по названию спектакля);

11. Вывести отсортированную (по названию спектакля) таблицу, используя упорядоченный массив ключей;

12. Вывести результаты сравнения эффективности программы (при обработке таблицы и массива ключей);

0. Выйти из программы.

**Обращение к программе:**

Запускается через терминал командой **make run**

Сообщения при аварийных ситуациях:

1. Не удалось прочитать кол-во записей в файле;
2. Недопустимое кол-во записей в файле;
3. Не удалось считать минимальную цену билета;
4. Не удалось считать максимальную цену билета;
5. Цена билета должна быть больше 0;
6. Минимальная цена билета не может превосходить максимальную;
7. Не удалось считать минимальный возраст;
8. Возраст не может быть отрицательным;
9. Возраст ребёнка не должен превышать 17 лет;

10) Введён неверный тип спектакля для детей;

11) В строке содержится избыточная информация;

12) Введён неверный тип спектакля для взрослых;

13) Ошибка в имени композитора;

14) Превышена максимальная длина имени композитора;

15) Ошибка в названии страны;

16) Превышена максимальная длина названия страны;

17) Минимальный возраст не может быть больше 23 лет (для посещения музыкального спектакля);

18) Не удалось считать продолжительность спектакля;

19) Продолжительность спектакля не может быть отрицательной;

20) Введён неверный тип спектакля;

21) Не удалось прочитать номер команды;

22) Номер команды должен быть >= 0 и <= 9;

23) Подходящих спектаклей не найдено.

**Описание структуры данных**

**table\_t** - структура, содержащая информацию о репертуарах театров, таблице ключей и её размере.

typedef struct

{

theaters\_t theaters[MAXIMUM\_TABLE\_SIZE];

keys\_t keys[MAXIMUM\_TABLE\_SIZE];

int size;

} table\_t;

**Поля структуры:**

1. theaters\_t theaters[MAXIMUM\_TABLE\_SIZE] – информация о репертуарах театров;
2. keys\_t keys[MAXIMUM\_TABLE\_SIZE] – информация о таблице ключей;
3. int size – размер таблицы.

MAXIMUM\_TABLE\_SIZE = 1000

**keys\_t** – структура, содержащая информацию о ключе и его индексе в основной таблице.

typedef struct

{

int index;

char array[MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME + 1];

} keys\_t;

**Поля структуры:**

1. int index – индекс элемента в основной таблице;
2. char array[MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME + 1] – ключ (название спектакля).

MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME = 20

**theaters\_t** – структура, содержащая информацию о репертуаре театра.

typedef struct

{

char theater\_name[MAXIMUM\_LENGTH\_OF\_THEATER\_NAME + 1];

char performance[MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME + 1];

char producer[MAXIMUM\_LENGTH\_OF\_PRODUCER\_NAME + 1];

int min\_price;

int max\_price;

char str\_type\_of\_performance[MAXIMUM\_LENGTH\_TYPE\_OF\_SPECTACL + 2];

type\_of\_performance\_t type\_of\_performance;

} theaters\_t;

**Поля структуры:**

1. char theater\_name[MAXIMUM\_LENGTH\_OF\_THEATER\_NAME + 1] – название театра;
2. char performance[MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME + 1] – название спектакля;
3. char producer[MAXIMUM\_LENGTH\_OF\_PRODUCER\_NAME + 1] – имя режиссёра;
4. int min\_price – минимальная цена билета;
5. int max\_price – максимальная цена билета;
6. char str\_type\_of\_performance[MAXIMUM\_LENGTH\_TYPE\_OF\_SPECTACL+2] – тип спектакля;
7. type\_of\_performance\_t type\_of\_performance – информация о спектаклях разных типов.

MAXIMUM\_LENGTH\_OF\_THEATER\_NAME = 23

MAXIMUM\_LENGTH\_OF\_PERFORMANCE\_NAME = 20

MAXIMUM\_LENGTH\_OF\_PRODUCER\_NAME = 20

MAXIMUM\_LENGTH\_TYPE\_OF\_SPECTACL = 7

**type\_of\_performance\_t** – объединение, содержащее информацию о спектаклях разных типов.

typedef union

{

child\_t child;

adult\_t adult;

musical\_t musical;

} type\_of\_performance\_t;

**Поля структуры:**

1. child\_t child - информация о детском спектакле;
2. adult\_t adult – информация о взрослом спектакле;
3. musical\_t musical – информация о музыкальном спектакле.

**musical\_t** – структура, содержащая информацию о музыкальном спектакле;

typedef struct

{

char composer[MAXIMUM\_COMPOSER\_NAME\_LENGTH + 1];

char country[MAXIMUM\_COUNTRY\_NAME\_LENGTH + 1];

int min\_age;

int duration;

} musical\_t;

**Поля структуры:**

1. char composer[MAXIMUM\_COMPOSER\_NAME\_LENGTH + 1] – имя композитора;
2. char country[MAXIMUM\_COUNTRY\_NAME\_LENGTH + 1] – название страны;
3. int min\_age – минимальный возраст для посещения спектакля;
4. int duration – продолжительность спектакля.

MAXIMUM\_COMPOSER\_NAME\_LENGTH = 20

MAXIMUM\_COUNTRY\_NAME\_LENGTH = 15

**adult\_t** – структура, содержащая информацию о спектакле для взрослых;

typedef struct

{

char type[MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_ADULTS + 2];

} adult\_t;

**Поля структуры:**

1. char type[MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_ADULTS + 2] – тип спектакля для взрослых (play, drama, comedy);

MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_ADULTS = 6

**child\_t** – структура, содержащая информацию о детском спектакле;

typedef struct

{

int min\_age;

char type[MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_CHILDREN + 2];

} child\_t;

**Поля структуры:**

1. int min\_age – минимальный возраст ребёнка для просмотра спектакля;
2. char type[MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_CHILDREN+2] - тип детского спектакля (story, play);

MAXIMUM\_LENGTH\_TYPE\_OF\_PERFOMANCE\_FOR\_CHILDREN = 5

# **Описание алгоритма**

1. Выводится меню программы (каждой команде присвоен номер);
2. Пользователь вводит номер команды, который отвечает за определённое действие;
3. Ввод осуществляется до того момента, пока не будет введён 0, являющийся признаком выхода из программы.

# **Набор тестов**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название теста** | **Входные данные** | **Результат** |
| 1 | Номер команды - число | k | Не удалось прочитать номер команды |
| 2 | Номер команды >= 0 и <=12 | 13 | Номер команды должен быть >= 0 и <= 9 |
| 3 | Неверно введено имя файла | dataaaaa.txt | Не удалось открыть файл |
| 4 | Первая строка файла не содержит кол-во строк таблицы | Первая строка файла не содержит кол-во строк таблицы | Не удалось прочитать кол-во записей в файле |
| 5 | В файле находятся недопустимые данные | Файл с некорректными данными | Вернётся сообщение, аналогичное сообщению при добавлении некорректной строки в таблицу. Также будет выведен номер строки, в которой была найдена ошибка |
| 6 | Невозможное количество записей в файле | Первая строка файла содержит число <= 0 или > 1000 | Недопустимое кол-во записей в файле |
| 7 | Количество возможных записей в таблице переполнено | Попытка добавить 1001 запись в конец таблицы | Нельзя добавить новый спектакль, так как таблица уже содержит максимальное кол-во строк |
| 8 | Недопустимый символ в названии театра (при добавлении нового спектакля в таблицу) | 9Concettina\_theater perfomance Merrin 126 1008 adult play | Встречен недопустимый символ '9' (Ошибка в названии театра) |
| 9 | Слишком длинное название театра (при добавлении нового спектакля в таблицу) | aaaaaaaConcettina\_theater perfomance Merrin 126 1008 adult play | Превышена максимальная длина названия театра |
| 10 | Недопустимый символ в названии спектакля (при добавлении нового спектакля в таблицу) | Concettina\_theater perfo\*mance Merrin 126 1008 adult play | Встречен недопустимый символ '\*' (Ошибка в названии спектакля) |
| 11 | Слишком длинное название спектакля (при добавлении нового спектакля в таблицу) | Concettina\_theater bbbbbbbbbbbperfomance Merrin 126 1008 adult play | Превышена максимальная длина названия спектакля |
| 12 | Недопустимый символ в имени режиссёра (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance \_Merrin 126 1008 adult play | Встречен недопустимый символ '\_' (Ошибка в имени режиссёра) |
| 13 | Слишком длинное имя режиссёра (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance Merrinccccccccccccccc 126 1008 adult play | Превышена максимальная длина имени режиссёра |
| 14 | Минимальная цена билета – число (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance Merrin k 1008 adult play | Не удалось считать минимальную цену билета |
| 15 | Максимальная цена билета – число (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance Merrin 126 k adult play | Не удалось считать максимальную цену билета |
| 16 | Цена билета – положительное число (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance Merrin -126 1008 adult play | Цена билета должна быть больше 0 |
| 17 | Минимальная цена билета больше минимальной (при добавлении нового спектакля в таблицу) | Concettina\_theater perfomance Merrin 1260 1008 adult play | Минимальная цена билета не может превосходить максимальную |
| 18 | Неверный тип спектакля для взрослых (при добавлении нового спектакля для взрослых в таблицу) | Concettina\_theater perfomance Merrin 126 1008 adult type | Введён неверный тип спектакля для взрослых |
| 19 | Минимальный возраст ребёнка – число (при добавлении нового детского спектакля в таблицу) | Concettina\_theater perfomance Merrin 126 1008 child k play | Не удалось считать минимальный возраст |
| 20 | Минимальный возраст ребёнка должен быть <18 (при добавлении нового детского спектакля в таблицу) | Concettina\_theater perfomance Merrin 126 1008 child 18 play | Возраст ребёнка не должен превышать 17 лет |
| 21 | Минимальный возраст ребёнка должен быть >=0 (при добавлении нового детского спектакля в таблицу) | Concettina\_theater perfomance Merrin 126 1008 child -18 play | Возраст не может быть отрицательным |
| 22 | Неверный тип детского спектакля (при добавлении нового детского спектакля в таблицу) | Concettina\_theater perfomance Merrin 126 1008 child 5 type | Введён неверный тип спектакля для детей |
| 23 | Недопустимый символ в имени композитора (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Gio?rgetti France 18 70 | Встречен недопустимый символ '?' (Ошибка в имени композитора) |
| 24 | Слишком длинное имя композитора (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical ooooooooooooGiorgetti France 18 70 | Превышена максимальная длина имени композитора |
| 25 | Недопустимый символ в названии страны (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France! 18 70 | Встречен недопустимый символ '!' (Ошибка в названии страны) |
| 26 | Слишком длинное название страны (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti Fraaaaaaaaaaaaance 18 70 | Превышена максимальная длина названия страны |
| 27 | Минимальный возраст – число (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France k 70 | Не удалось считать минимальный возраст |
| 28 | Минимальный возраст должен быть >=0 (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France -4 70 | Возраст должен быть положительным числом |
| 29 | Минимальный возраст должен быть <24 (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France 24 70 | Минимальный возраст не может быть больше 23 лет (для посещения музыкального спектакля) |
| 30 | Продолжительность спектакля - число (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France 18 k | Не удалось считать продолжительность спектакля |
| 31 | Продолжительность спектакля должна быть >0 (при добавлении нового музыкального спектакля в таблицу) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France 18 0 | Продолжительность спектакля должна быть больше 0 |
| 32 | Цена – число (Удаление спектаклей, минимальная цена билетов которых больше указанного значения) | k | Не удалось считать верхнюю границу минимальной цены билета |
| 33 | Цена > 0 (Удаление спектаклей, минимальная цена билетов которых больше указанного значения) | -3 | Верхняя граница минимальной цены билета должна быть больше 0 |
| 34 | Минимальный возраст – число (Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной) | k | Не удалось считать минимальный возраст |
| 35 | Минимальный возраст <=23 (Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной) | 24 | Минимальный возраст не может быть больше 23 |
| 36 | Минимальный возраст >=0 (Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной) | -3 | Возраст не может быть отрицательным |
| 37 | Продолжительность спектакля - число (Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной) | 5  k | Не удалось считать продолжительность спектакля |
| 38 | Продолжительность спектакля >=0 (Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной) | 5  -3 | Продолжительность спектакля не может быть отрицательной |
| 39 | Добавление верной записи в конец таблицы (детский спектакль (story)) | Maia\_theater perfomance Tourot 467 912 child 16 story | Спектакль был успешно добавлен |
| 40 | Добавление верной записи в конец таблицы (детский спектакль (play)) | Maia\_theater perfomance Tourot 467 912 child 16 play | Спектакль был успешно добавлен |
| 41 | Добавление верной записи в конец таблицы (спектакль для взрослых (play)) | Keely\_theater perfomance Mithon 318 1040 adult drama | Спектакль был успешно добавлен |
| 42 | Добавление верной записи в конец таблицы (спектакль для взрослых (drama)) | Keely\_theater perfomance Mithon 318 1040 adult drama | Спектакль был успешно добавлен |
| 43 | Добавление верной записи в конец таблицы (спектакль для взрослых (comedy)) | Keely\_theater perfomance Mithon 318 1040 adult comedy | Спектакль был успешно добавлен |
| 44 | Добавление верной записи в конец таблицы (музыкальный спектакль) | Maia\_theater perfomance Merrin 126 1008 musical Giorgetti France 18 70 | Спектакль был успешно добавлен |
| 45 | Загрузка списка спектаклей из файла в таблицу | Команда 1 | Заполненная таблица |
| 46 | Удаление спектаклей, минимальная цена билетов которых больше указанного значения | 200  (в таблице есть спектакли, которые нужно удалить) | Кол-во удалённых спектаклей - 72 |
| 47 | Удаление спектаклей, минимальная цена билетов которых больше указанного значения | 500  (в таблице нет спектаклей, которые нужно удалить) | Кол-во удалённых спектаклей - 0 |
| 48 | Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной (такие записи есть) | 5  120 | Выведется таблица с нужными спектаклями и сообщение с их кол-вом (Кол-во подходящих спектаклей – 25) |
| 49 | Вывести список всех музыкальных спектаклей для детей указанного возраста с продолжительностью меньше указанной (таких записей нет) | 5  50 | Подходящих спектаклей не найдено |
| 50 | Вывод таблицы (таблица не пуста) | Команда 2 | Успешный вывод таблицы |
| 51 | Вывод таблицы (таблица пуста) | Команда 2 | Таблица пуста |
| 52 | Вывод таблицы ключей (таблица не пуста) | Команда 6 | Успешный вывод таблицы ключей |
| 53 | Вывод таблицы ключей (таблица пуста) | Команда 6 | Таблица пуста |
| 54 | Сортировки массива ключей  (массив не пустой) | Команды 7,8 | Массив ключей успешно отсортирован |
| 55 | Сортировки таблицы  (таблица не пуста) | Команды 9,10 | Таблица успешно отсортирована |
| 56 | Сортировки таблицы  (таблица пуста) | Команды 9,10,11 | Таблица пуста |
| 57 | Вывод отсортированной (по названию спектакля) таблицы, используя упорядоченный массив ключей (таблица не пуста) | Команда 11 | Вывод отсортированной таблицы |
| 58 | Вывод результатов сравнения эффективности программы (при обработке таблицы и массива ключей) | Команда 12 | Вывод результатов сравнения сортировок |
| 59 | Выход из программы | Команда 0 | Выход из программы |

# **Оценка эффективности**

Сортировка каждой таблицы будет измеряться в тактах процессора (процессор со средней частотой 2.3gHz).

**Время сортировки:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Количество записей | Полная таблица | | Таблица ключей | |
| Сортировка «выбором» | Сортировка «qsort» | Сортировка «выбором» | Сортировка «qsort» |
| 60 | 25760 | 7858 | 20991 | 6733 |
| 120 | 100494 | 22166 | 88008 | 14866 |
| 240 | 352799 | 60061 | 346903 | 45751 |
| 480 | 1322370 | 138691 | 1308409 | 124770 |

**Объем занимаемой памяти:**

|  |  |  |
| --- | --- | --- |
| Количество записей | Полная таблица | Таблица ключей |
| 60 | 8160 | 1680 |
| 120 | 16320 | 3360 |
| 240 | 32640 | 6720 |
| 480 | 65280 | 13440 |

**Таблица соотношений памяти и времени:**

|  |  |  |  |
| --- | --- | --- | --- |
| Количество записей | % памяти, занимаемый таблицей ключей от всей таблицы | На сколько % сортировка таблицы ключей быстрее сортировки всей таблицы (“выбором ”) | На сколько % сортировка массива ключей быстрее сортировки всей таблицы (“qsort”) |
| 60 | 21% | 19% | 14% |
| 120 | 21% | 12% | 33% |
| 240 | 21% | 2% | 24% |
| 480 | 21% | 1% | 10% |

# **Ответы на контрольные вопросы**

1.Как выделяется память под вариантную часть записи?

Максимальное по длине поле вариантной части - размер памяти, который выделяется под вариантную часть. Эта память является общей для всех полей вариантной части записи.

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

В вариантной части при компиляции тип данных не проверяется. Невозможно считать корректно данные, поэтому поведение будет неопределенным.

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

За правильностью выполнения операций с вариантной частью записи должен следить программист.

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

Таблица ключей представляет собой структуры из двух массивов, один под индексы, другой под поля для хранения. Использование таблицы ключей при сортировке, ускоряет ее работу, но требует дополнительной памяти.

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

Если исходная таблица содержит небольшое количество полей, то лучше сортировать именно её. Но если в структуре большое количество полей, то стоит создать дополнительную таблицу ключей, которая займет дополнительную память, но ускорит работу сортировки.

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

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

# **Вывод**

В процессе выполнения данной лабораторной работы был приобретен опыт работы со структурами с вариантной частью и обработки таблицы данных.

Использование таблицы ключей помогает оптимизировать сортировку таблицы с большим количеством полей, что даёт хороший прирост (до 29% методом “выбором” и до 30% методом “qsort”) во времени, но требует дополнительной памяти (21%). Стоит отметить, что использование массива ключей неэффективно при небольших размерах самой таблицы. В таком случае эффективнее просто отсортировать таблицу, так как разница во времени будет несущественна, а затраты на память сократятся.

Также не стоит игнорировать важность выбора метода сортировки. Не стоит выбирать сортировки сложности О(n^2) для обработки таблиц ключей больших объёмов, так как вы не получите большого прироста во времени (1%), но сильно потеряете в памяти (21%).

Значение ключа играет немаловажную роль для быстроты сортировки. Сортировка по слову требует намного больше времени и памяти, чем по числу. Так же сортировка таблицы ключей (ключ - слово) при больших объёмах данных менее эффективна, чем при малых (метод сортировки “выбором” хорошо это показывает).