

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский Авиационный Институт»
(Национальный Исследовательский Университет)

Институт: №8 «Информационные технологии и прикладная
математика»
Кафедра: 806 «Вычислительная математика и программирование»

Курсовая работа
по курсу «Практикум на ЭВМ»
II семестр
Задание 9
«Сортировка и поиск»

Группа	М8О-107Б-20
Студент	Чекменев В.А.
Преподаватель	Найдёнов И.Е.
Оценка	
Дата	

Постановка задачи

Составить программу на языке Си с использованием процедур и функций для сортировки таблицы заданным методом и двоичного поиска по ключу в таблице.

Тип ключа: целое, 8 байт (long int).

Метод сортировки: шейкер сортировка.

Способ хранения: вместе.

Теория

шейкер сортировка — разновидность пузырьковой сортировки.

Сортировка **пузырьком** — это **метод** сортировки массивов и списков путем последовательного сравнения и обмена соседних элементов, если предшествующий оказывается больше последующего.

Анализируя метод пузырьковой сортировки, можно отметить два обстоятельства.

Во-первых, если при движении по части массива перестановки не происходят, то эта часть массива уже отсортирована и, следовательно, её можно исключить из рассмотрения.

Во-вторых, при движении от конца массива к началу минимальный элемент «всплывает» на первую позицию, а максимальный элемент сдвигается только на одну позицию вправо.

Эти две идеи приводят к следующим модификациям в методе пузырьковой сортировки. Границы рабочей части массива (то есть части массива, где происходит движение) устанавливаются в месте последнего обмена на каждой итерации. Массив просматривается поочередно справа налево и слева направо.

Двоичный (бинарный) поиск (также известен как **метод деления пополам** или **дихотомия**) — классический алгоритм поиска элемента в отсортированном массиве (векторе), использующий дробление массива на половины.

Описание алгоритма

В программе используются классические, всем известные алгоритмы.

Описывать их работу считаю излишним.

Бинарный поиск - сложность $O(\log N)$

шейкер сортировка — сложность:

Лучший случай $O(n)$

Средний случай $O(n^2)$

Наихудший случай $O(n^2)$

Перемешивание и реверс – сложность $O(N)$.

Данные хранятся в виде структуры row, где, согласно условию задачи, лежат ключ со значением – целое число и строка соответственно.

Считывание происходит построчно – берём из файла сначала целое число, а затем строку, пока не встретим символ перевода строки или пока не закончится место в массиве (если входная строка слишком длинная).

Листинг программ

table.h – заголовочный файл с функциями для обработки таблицы

```
#ifndef TABLE_H
#define TABLE_H

#include <stdio.h>

typedef struct _row{
    long int _key;
    char _str[120];
} row;

void printTable(row *arr, const int size); // Печать таблицы
void getRow(FILE *stream, char *str, const int size); // Получить одну строку
из файла
void swapRows(row *r1, row *r2); // swap for row

#endif
```

table.c – функции для обработки таблицы

```
#include "table.h"
#include <stdio.h>

void printTable(row *arr, const int size)
{
    printf("|   Ключ   |               строка               |\n");
    for (int i = 0; i < size; i++) {
```

```

        printf("| %9ld | %50s | \n", arr[i]._key, arr[i]._str);
    }
}

void getRow(FILE *stream, char *str, const int size)
{
    int cnt = 0;
    char ch;
    while ((ch = getc(stream)) != '\n' && cnt < size - 1) {
        str[cnt++] = ch;
    }
    str[cnt] = '\0';
}

void swapRows(row *r1, row *r2)
{
    row tmp;
    tmp = *r1;
    *r1 = *r2;
    *r2 = tmp;
}

```

func.h – заголовочный файл с основными функциями и функцией сортировки

```

#ifndef FUNC_H
#define FUNC_H

#include "table.h"

int binSearch(const row *mass, const int size, const long int key); // Бинарный
поиск в таблице по ключу
void shakerSort(row *mass, const int count); // Сортировка шейкером
void shuffle(row *mass, const int size); // Перемешивание данных в таблице
void reverse(row *mass, const int size); // Реверс данных в таблице
int randomAB(const int a, const int b);
int isSorted(const row *mass, const int size); // Проверка на
отсортированность таблицы

#endif

```

func.c – основные функции и функция сортировки шейкером

```
#include <time.h>
#include <stdlib.h>
#include "table.h"
#include <stdio.h>

int binSearch(const row *mass, const int size, const long int key)
{
    int start = 0, end = size - 1, mid;

    if (size <= 0)
        return -1;

    while (start < end) {
        mid = start + (end - start) / 2;
        if (mass[mid]._key == key) {
            return mid;
        } else if (mass[mid]._key < key) {
            start = mid + 1;
        } else {
            end = mid;
        }
    }
    if (mass[end]._key == key) {
        return end;
    }
    return -1;
}

// Функция Шейкер-сортировки
void shakerSort(row *mass, const int count)
{
    int left = 0, right = count - 1;
    int flag = 1;

    while ((left < right) && flag > 0) {
        flag = 0;

        for (int i = left; i < right; i++) {
            if (mass[i]._key > mass[i + 1]._key) {
                swapRows(&mass[i], &mass[i + 1]);
                flag = 1;
            }
        }
    }
}
```

```

    }
}
right--;

for (int i = right; i > left; i--) {
    if (mass[i - 1]._key > mass[i]._key) {
        swapRows(&mass[i], &mass[i-1]);
        flag = 1;
    }
}
left++;
}
}

int randomAB(const int a, const int b)
{
    return a + rand() % (b - a + 1);
}

void shuffle(row *mass, const int size)
{
    int i, j, k;

    srand((unsigned int)time(0));

    for (k = 0; k < size; k++) {
        i = randomAB(0, size - 1);
        j = randomAB(0, size - 1);
        swapRows(&mass[i], &mass[j]);
    }
}

void reverse(row *mass, const int size)
{
    int i, j;
    for (i = 0, j = size - 1; i < j; i++, j--) {
        swapRows(&mass[i], &mass[j]);
    }
}

int isSorted(const row *mass, const int size)
{
    for (int i = 0; i < size - 1; i++) {

```

```

        if (mass[i]._key > mass[i + 1]._key) {
            return 0;
        }
    }
    return 1;
}

```

client.h – интерфейс

```

#include <stdio.h>
#include <stdlib.h>
#include "func.h"

```

```

int main(void)
{
    printf("Напишите '?' для получения помощи в использовании
программы:\n");
    char c;

    const int N = 50;
    int i, cnt, action;
    long key;
    char ch;
    row mass[N];
    FILE *file = fopen("input.txt", "r");

    if (file == NULL) {
        printf("Ошибка при открытии файла\n");
        return 0;
    }
    i = 0;
    while (i < N && fscanf(file, "%ld", &mass[i]._key) == 1) {
        fscanf(file, "%c", &ch);
        getRow(file, mass[i]._str, sizeof(mass[i]._str));
        i++;
    }
    fclose(file);
    cnt = i;

    while ((c = getchar()) != EOF) {
        if (c == '?') {
            printf("Набор команд:\n");

```

```

printf("1 - вывести таблицу.\n");
printf("2 - найти и вывести в таблице строку по ключу.\n");
printf("3 - отсортировать таблицу по ключам.\n");
printf("4 - перемешать строки таблицы.\n");
printf("5 - перевести таблицу в обратный порядок.\n");
printf("6 - закончить сеанс.\n");
} else if (c == '1') {
    printTable(mass, cnt);
} else if (c == '2') {
    if (!isSorted(mass, cnt)) {
        printf("Таблица не отсортирована...\n");
    } else {
        printf("Введите ключик: ");
        scanf("%ld", &key);
        i = binSearch(mass, cnt, key);
        if (i > -1) {
            printf("строка по ключу: %s\n", mass[i]._str);
        } else {
            printf("Элементов с таким ключом не существует!\n");
        }
    }
}
} else if (c == '3') {
    shakerSort(mass, cnt);
} else if (c == '4') {
    shuffle(mass, cnt);
} else if (c == '5') {
    reverse(mass, cnt);
} else if (c == '6') {
    printf("все на сегодня...\n");
    return 0;
} else if (c != '?' && c != '1' && c != '2' && c != '3' && c != '4' && c != '5'
&& c != '6' && c != '\n' && c != '\t' && c != ' ') {
    printf("Не та буква, попробуйте еще раз...\n");
}
}
return 0;
}

```


Листинг терминала во время тестирования программы

файл test.txt содержит строки стиха Роберта Ли Фроста.

тест 1 – вывод таблицы

```
[suraba04@asusx512fl cp9]$ gcc *.c -o test0
```

```
[suraba04@asusx512fl cp9]$ ./test0
```

Напишите '?' для получения помощи в использовании программы:

?

Набор команд:

- 1 - вывести таблицу.
- 2 - найти и вывести в таблице строку по ключу.
- 3 - отсортировать таблицу по ключам.
- 4 - перемешать строки таблицы.
- 5 - перевести таблицу в обратный порядок.
- 6 - закончить сеанс.

1

Ключ	строка	
1	Two roads diverged in a yellow wood,	
2	And sorry I could not travel both	
3	And be one traveler, long I stood	
4	And looked down one as far as I could	
5	To where it bent in the undergrowth.	
6	Then took the other, as just as fair,	
7	And having perhaps the better claim,	
8	Because it was grassy and wanted wear;	
9	something...	
10	Though as for that the passing there	
11	Had worn them really about the same.	
12	And both that morning equally lay	

13	In leaves no step had trodden black.
14	Oh, I kept the first for another day!
15	Yet knowing how way leads on to way,
16	I doubted if I should ever come back.
17	I shall be telling this with a sigh
18	Somewhere ages and ages hence:
19	Two roads diverged in a wood, and I—
20	I took the one less traveled by,
21	And that has made all the difference.

6

все на сегодня...

тест 2.1 – найти и вывести в таблице строку по ключу

[suraba04@asusx512fl cp9]\$./test0

Напишите '?' для получения помощи в использовании программы:

,

Не та буковка, попробуйте еще раз...

?

Набор команд:

1 - вывести таблицу.

2 - найти и вывести в таблице строку по ключу.

3 - отсортировать таблицу по ключам.

4 - перемешать строки таблицы.

5 - перевести таблицу в обратный порядок.

6 - закончить сеанс.

2

Введите ключик: 5

строка по ключу: To where it bent in the undergrowth.

6

все на сегодня...

тест 2.2 – найти и вывести в таблице строку по ключу, неверный ключ

```
[suraba04@asusx512fl cp9]$ ./test0
```

Напишите '?' для получения помощи в использовании программы:

?

Набор команд:

- 1 - вывести таблицу.
- 2 - найти и вывести в таблице строку по ключу.
- 3 - отсортировать таблицу по ключам.
- 4 - перемешать строки таблицы.
- 5 - перевести таблицу в обратный порядок.
- 6 - закончить сеанс.

2

Введите ключик: 46

Элементов с таким ключом не существует!

6

все на сегодня...

тест 3 – отсортировать таблицу по ключам

```
[suraba04@asusx512fl cp9]$ ./test0
```

Напишите '?' для получения помощи в использовании программы:

?

Набор команд:

- 1 - вывести таблицу.

2 - найти и вывести в таблице строку по ключу.

3 - отсортировать таблицу по ключам.

4 - перемешать строки таблицы.

5 - перевести таблицу в обратный порядок.

6 - закончить сеанс.

4

1

Ключ	строка	
10	Though as for that the passing there	
8	Because it was grassy and wanted wear;	
3	And be one traveler, long I stood	
15	Yet knowing how way leads on to way,	
5	To where it bent in the undergrowth.	
20	I took the one less traveled by,	
18	Somewhere ages and ages hence:	
19	Two roads diverged in a wood, and I—	
1	Two roads diverged in a yellow wood,	
6	Then took the other, as just as fair,	
21	And that has made all the difference.	
11	Had worn them really about the same.	
12	And both that morning equally lay	
4	And looked down one as far as I could	
9	something...	
7	And having perhaps the better claim,	
2	And sorry I could not travel both	
13	In leaves no step had trodden black.	
14	Oh, I kept the first for another day!	

| 17| I shall be telling this with a sigh|
| 16| I doubted if I should ever come back.|

3

1

Ключ	строка
1	Two roads diverged in a yellow wood,
2	And sorry I could not travel both
3	And be one traveler, long I stood
4	And looked down one as far as I could
5	To where it bent in the undergrowth.
6	Then took the other, as just as fair,
7	And having perhaps the better claim,
8	Because it was grassy and wanted wear;
9	something...
10	Though as for that the passing there
11	Had worn them really about the same.
12	And both that morning equally lay
13	In leaves no step had trodden black.
14	Oh, I kept the first for another day!
15	Yet knowing how way leads on to way,
16	I doubted if I should ever come back.
17	I shall be telling this with a sigh
18	Somewhere ages and ages hence:
19	Two roads diverged in a wood, and I—
20	I took the one less traveled by,
21	And that has made all the difference.

6

все на сегодня...

тест 4 – перемешать строки таблицы и вывести

[suraba04@asusx512fl cp9]\$./test0

Напишите '?' для получения помощи в использовании программы:

?

Набор команд:

1 - вывести таблицу.

2 - найти и вывести в таблице строку по ключу.

3 - отсортировать таблицу по ключам.

4 - перемешать строки таблицы.

5 - перевести таблицу в обратный порядок.

6 - закончить сеанс.

4

1

Ключ	строка	
10	Though as for that the passing there	
9	something...	
3	And be one traveler, long I stood	
4	And looked down one as far as I could	
6	Then took the other, as just as fair,	
12	And both that morning equally lay	
14	Oh, I kept the first for another day!	
13	In leaves no step had trodden black.	
16	I doubted if I should ever come back.	
18	Somewhere ages and ages hence:	

15	Yet knowing how way leads on to way,
11	Had worn them really about the same.
2	And sorry I could not travel both
19	Two roads diverged in a wood, and I—
8	Because it was grassy and wanted wear;
5	To where it bent in the undergrowth.
1	Two roads diverged in a yellow wood,
7	And having perhaps the better claim,
17	I shall be telling this with a sigh
20	I took the one less traveled by,
21	And that has made all the difference.

6

все на сегодня...

Тест 5 – перевести таблицу в обратный порядок и вывести

[suraba04@asusx512fl cp9]\$./test0

Напишите '?' для получения помощи в использовании программы:

?

Набор команд:

1 - вывести таблицу.

2 - найти и вывести в таблице строку по ключу.

3 - отсортировать таблицу по ключам.

4 - перемешать строки таблицы.

5 - перевести таблицу в обратный порядок.

6 - закончить сеанс.

3

Ключ	строка	
1	Two roads diverged in a yellow wood,	
2	And sorry I could not travel both	
3	And be one traveler, long I stood	
4	And looked down one as far as I could	
5	To where it bent in the undergrowth.	
6	Then took the other, as just as fair,	
7	And having perhaps the better claim,	
8	Because it was grassy and wanted wear;	
9	something...	
10	Though as for that the passing there	
11	Had worn them really about the same.	
12	And both that morning equally lay	
13	In leaves no step had trodden black.	
14	Oh, I kept the first for another day!	
15	Yet knowing how way leads on to way,	
16	I doubted if I should ever come back.	
17	I shall be telling this with a sigh	
18	Somewhere ages and ages hence:	
19	Two roads diverged in a wood, and I—	
20	I took the one less traveled by,	
21	And that has made all the difference.	

1

	Ключ		строка	
	21		And that has made all the difference.	
	20		I took the one less traveled by,	
	19		Two roads diverged in a wood, and I—	
	18		Somewhere ages and ages hence:	
	17		I shall be telling this with a sigh	
	16		I doubted if I should ever come back.	
	15		Yet knowing how way leads on to way,	
	14		Oh, I kept the first for another day!	
	13		In leaves no step had trodden black.	
	12		And both that morning equally lay	
	11		Had worn them really about the same.	
	10		Though as for that the passing there	
	9		something...	
	8		Because it was grassy and wanted wear;	
	7		And having perhaps the better claim,	
	6		Then took the other, as just as fair,	
	5		To where it bent in the undergrowth.	
	4		And looked down one as far as I could	
	3		And be one traveler, long I stood	
	2		And sorry I could not travel both	
	1		Two roads diverged in a yellow wood,	

6

все на сегодня...

Вывод

В данной курсовой работе я научился работать с шейкер-сортировкой и с бинарным поиском. Больше всего понравилось писать сами алгоритмы. Самой сложной частью в работе было отлавливание крайних случаев. Думаю, что в будущем такая тема как алгоритмы очень сильно мне поможет, так как это база всех программ в совокупности с структурами данных.