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	Алапанова Э.Х.
Преподаватель	Найдёнов И.Е.
Оценка	
Дата	

Москва, 2021

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: целый, 4 байт.
Метод сортировки: пирамидальная сортировка с просеиванием.
Способ хранения: отдельно.

Теория

Пирамидальная сортировка (или сортировка кучей, HeapSort) — это метод сортировки сравнением, основанный на такой структуре данных как двоичная куча. Она похожа на сортировку выбором, где мы сначала ищем максимальный элемент и помещаем его в конец. Далее мы повторяем ту же операцию для оставшихся элементов.

Исходной код

Файл main.c

```
#include "KP9.h"
int main(void)
{
	const int N = 50;
	int i, cnt, action;
	char ch;
	Key keys[N];
	Val values[N];
	Key key;
	FILE *file = fopen("input.txt", "r");

	if (file == NULL)
	{
		printf("Ошибка при открытии файла\n");

		return 0;
	}

	i = 0;

	while (i < N && fscanf(file, "%d", &keys[i].key) == 1)
	{
```

```
	fscanf(file, "%c", &ch);
	getRow(file, values[i].val, sizeof(values[i].val));

	i++;
}

fclose(file);

cnt = i;

do
{
	printf("Меню\n");
	printf("1) Печать\n");
	printf("2) Двоичный поиск\n");
	printf("3) Сортировка\n");
	printf("4) Перемешивание\n");
	printf("5) Реверс\n");
	printf("6) Выход\n");
	printf("Выберите действие\n");
	scanf("%d", &action);

	switch (action)
	{
		case 1:
		{
			printTable(keys, values, cnt);

			break;
		}

		case 2:
		{
			if (!isSorted(keys, cnt))
				printf("Ошибка. Таблица не отсортирована\n");
			else
			{
				printf("Введите ключ: ");
				scanf("%d", &key.key);

				i = binSearch(keys, values, cnt, key);

				if (i > -1)
```

```
                                    printf("Найдена строка: %s\n",
values[i].val);
                              else
                                    printf("Строка с таким ключом не
найдена\n");
                        }

                        break;
                  }

                  case 3:
                  {
                        sort(keys, values, cnt);

                        break;
                  }

                  case 4:
                  {
                        scramble(keys, values, cnt);

                        break;
                  }

                  case 5:
                  {
                        reverse(keys, values, cnt);

                        break;
                  }

                  case 6: break;

                  default:
                  {
                        printf("Ошибка. Такого пункта меню не существует\
n");

                        break;
                  }
            }
      }
      while (action != 6);
```

```
	return 0;
}
```

Файл kp9.h

```
#ifndef KP9_H
#define KP9_H
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>



typedef struct _Key
{
		int key;
} Key;


typedef struct _Val
{
		char val[81];
} Val;


void printTable(const Key* k, const Val* v, const int size);
int binSearch(const Key* k, const Val* v, const int size, const Key key);
void sort(Key* k, Val* v, const int size);
void scramble(Key* k, Val* v, const int size);
void reverse(Key* k, Val* v, const int size);
void swap(Key* k1, Key* k2, Val* v1, Val* v2);
void sift(Key* k, Val* v, int start, int end);



void getRow(FILE* stream, char* str, const int size);
void swapRows(Key* k, Val* v, const int a, const int b);
int comparator(const Key k1, const Key k2);
int isEqualKeys(const Key k1, const Key k2);
int randomAB(const int a, const int b);
int isSorted(const Key* k, const int size);




#endif
```

Файл kp9.c

```
#include "KP9.h"


void printTable(const Key* k, const Val* v, const int size)
{
		int i;
```

```
	printf("+--------+------------------------------------------------------------+\n");
	printf("|  Ключ |                    Значение                                |\n");
	printf("+--------+------------------------------------------------------------+\n");

	for (i = 0; i < size; i++)
		printf("|%8d|%60s|\n", k[i].key, v[i].val);

	printf("+--------+------------------------------------------------------------+\n");
}

int binSearch(const Key* k, const Val* v, const int size, const Key key)
{
	int start = 0, end = size - 1, mid;

	if (size <= 0)
		return -1;

	while (start < end)
	{
		mid = start + (end - start) / 2;

		if (isEqualKeys(k[mid], key))
			return mid;
		else if (comparator(k[mid], key))
			start = mid + 1;
		else
			end = mid;
	}

	if (isEqualKeys(k[end], key))
		return end;

	return -1;
}

void swap(Key* k1, Key* k2, Val* v1, Val* v2)
{
	Key t;
	Val f;
	t = *k1;
	*k1 = *k2;
	*k2 = t;
	f = *v1;
	*v1 = *v2;
	*v2 = f;
}

void sort(Key* k, Val* v, const int size)
{
	int i;

	for (i = (size - 2) / 2; i >= 0; i--)
		sift(k, v, i, size);

	for (i = size - 1; i > 0; i--)
```

```
		{
			swap(&k[0], &k[i], &v[0], &v[i]);
			sift(k, v, 0, i);
		}
}

void sift(Key* k, Val* v, int start, int end)
{
		int root = start;
		int child = root * 2 + 1;

		while (child < end)
		{
			if (child + 1 < end && k[child].key < k[child + 1].key)
				child++;

			if (k[root].key >= k[child].key)
				break;

			swap(&k[root], &k[child], &v[root], &v[child]);

			root = child;
			child = root * 2 + 1;
		}
}


void scramble(Key* k, Val* v, const int size)
{
		int i, j, z;

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

		for (z = 0; z < size; z++)
		{
			i = randomAB(0, size - 1);
			j = randomAB(0, size - 1);

			swapRows(k, v, i, j);
		}
}

void reverse(Key* k, Val* v, const int size)
{
		int i, j;

		for (i = 0, j = size - 1; i < j; i++, j--)
			swapRows(k, v, i, j);
}

void getRow(FILE* stream, char* str, const int size)
{
		int cnt = 0, ch;

		while ((ch = getc(stream)) != '\n' && cnt < size - 1)
```

```
                        str[cnt++] = ch;

        str[cnt] = '\0';
}

void swapRows(Key* k, Val* v, const int a, const int b)
{
        Key tmpKey;
        Val tmpVal;

        tmpKey = k[a];
        k[a] = k[b];
        k[b] = tmpKey;

        tmpVal = v[a];
        v[a] = v[b];
        v[b] = tmpVal;
}

int comparator(const Key k1, const Key k2)
{
        return k2.key >= k1.key;
}

int isEqualKeys(const Key k1, const Key k2)
{
        return k1.key == k2.key;
}

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

int isSorted(const Key* k, const int size)
{
        int i;

        for (i = 0; i < size - 1; i++)
                if (!comparator(k[i], k[i + 1]))
                        return 0;

        return 1;
}
```

Протокол исполнения и тесты

```
elza@elza-NBLB-WAX9N:~/kp9$ gcc KP9.c main.c
elza@elza-NBLB-WAX9N:~/kp9$ ./a.out
Меню
1) Печать
2) Двоичный поиск
3) Сортировка
4) Перемешивание
5) Реверс
6) Выход
Выберите действие
```

Входной текст : input.txt

Тест №1. Печать (вывод таблицы в терминале выглядит лучше, чем в word)

Меню

1) Печать

2) Двоичный поиск

3) Сортировка

4) Перемешивание

5) Реверс

6) Выход

Выберите действие

1

Ключ	Значение
1	I'm waking up to ash and dust
2	I wipe my brow and sweat my rust
3	I'm breathing in the chemicals
4	I'm breaking in and shaping up
5	Then checking out on the prison bus
6	This is it the apocalypse
7	I'm waking up
8	I feel it in my bones
9	Enough to make my system blow
10	Welcome to the new age
11	To the new age
12	Welcome to the new age

```
|      13|                                To the new age|
|      14|                               I'm radioactive|
|      15|                                  Radioactive|
|      16|                               I'm radioactive|
|      17|                                  Radioactive|
+--------+---------------------------------------------------------+
```

Тест №2. Реверс таблицы

Меню

1) Печать

2) Двоичный поиск

3) Сортировка

4) Перемешивание

5) Реверс

6) Выход

Выберите действие

1

```
+--------+---------------------------------------------------------+
|  Ключ  |                         Значение                        |
+--------+---------------------------------------------------------+
|      17|                                  Radioactive|
|      16|                               I'm radioactive|
|      15|                                  Radioactive|
|      14|                               I'm radioactive|
|      13|                                To the new age|
|      12|                       Welcome to the new age|
|      11|                                To the new age|
```

10	Welcome to the new age
9	Enough to make my system blow
8	I feel it in my bones
7	I'm waking up
6	This is it the apocalypse
5	Then checking out on the prison bus
4	I'm breaking in and shaping up
3	I'm breathing in the chemicals
2	I wipe my brow and sweat my rust
1	I'm waking up to ash and dust

Тест №3. Перемешать таблицу

Ключ	Значение
2	I wipe my brow and sweat my rust
16	I'm radioactive
7	I'm waking up
15	Radioactive
4	I'm breaking in and shaping up
11	To the new age
13	To the new age
5	Then checking out on the prison bus
10	Welcome to the new age
17	Radioactive
6	This is it the apocalypse

| 8| I feel it in my bones|

| 1| I'm waking up to ash and dust|

| 14| I'm radioactive|

| 3| I'm breathing in the chemicals|

| 12| Welcome to the new age|

| 9| Enough to make my system blow|

+--------+----------+

Тест №4. Воспользуемся бинарным поиском на неотсортированной таблице

Меню

1) Печать

2) Двоичный поиск

3) Сортировка

4) Перемешивание

5) Реверс

6) Выход

Выберите действие

2

Ошибка. Таблица не отсортирована

Тест №5. Отсортируем таблицу (исходная таблица – результат теста 3)

+--------+----------+

| Ключ | Значение |

+--------+----------+

| 1| I'm waking up to ash and dust|

| 2| I wipe my brow and sweat my rust|

| 3| I'm breathing in the chemicals|

4	I'm breaking in and shaping up
5	Then checking out on the prison bus
6	This is it the apocalypse
7	I'm waking up
8	I feel it in my bones
9	Enough to make my system blow
10	Welcome to the new age
11	To the new age
12	Welcome to the new age
13	To the new age
14	I'm radioactive
15	Radioactive
16	I'm radioactive
17	Radioactive

+--------+----------+

Тест №6. Бинарный поиск.

Меню

1) Печать

2) Двоичный поиск

3) Сортировка

4) Перемешивание

5) Реверс

6) Выход

Выберите действие

2

Введите ключ: 6

Найдена строка: This is it the apocalypse

Вывод

Хорошая курсовая работа. Понравилось, что заставили изучить классические алгоритмы.

Примечание: ссылка на GitHub с кодом программы https://github.com/alpnva/MAI/tree/main/kp9