|  |  |  |
| --- | --- | --- |
|  | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | |
|  | | |
| Кафедра прикладной математики | | |
| Практическое задание № 4 | | |
| по дисциплине «Структуры данных и алгоритмы» | | |
| **Управление таблицами** | | |
|  | | |
|  | Бригада 6 | Сосов Егор |
| Группа ПМ-15 | Шестаков Алексей |
| Вариант 2б | Микаилов Фуад |
|  |  |
|  |  |
|  |  |
| Преподаватель | Тракимус Юрий Викторович |
|
| Новосибирск, 2022 | | |

1. **Условие задачи**

В файле WORK содержатся результаты работы цеха за день. Элемент файла включает: шифр изделия ( 8 - символьный код ), наименование изделия, количество (штук). Построить таблицу, содержащую результаты работы за день, считая ключом шифр изделия. Элемент таблицы имеет ту же структуру, что и элемент файла. Содержащаяся в файле информация с равными ключами должны быть помещена в таблицу один раз с общим количеством штук изделия. Организовать таблицу как:

б) упорядоченную;

1. **Анализ задачи   
   2.1. Входные и выходные данные** Входные данные: файл, элемент которого содержит 8-символьный код изделия, наименование изделия и количество штук через пробел.

Выходные данные: файл, элемент которого содержит 8-символьный код изделия, наименование изделия и количество штук через пробел.

**2.2. Решение задачи**

1. С помощью подпрограмм «input», «insert», «binarySearch», «shift» строим и заполняем таблицу из файла.

2. Запрашиваем индекс элемент таблица, который хотим удалить.

3. С помощью подпрограммы «binarySearch» проверяем наличие элемента в таблице и удаляем подпрограммой «deleteF» элемент таблицы по ключу.

4. Записываем данные таблицы в файл подпрограммой «output».

1. **Структура данных**

Структура «таблица» представляется упорядоченным массивом и счетчиком элементов таблицы. Элемент таблицы содержит три информационных поля: key, name – массивы символов типа char, quantity – элемент типа UINT.

key

name

quantity

1. **Текст программы**

#include <stdio.h>

#include <conio.h>

#include <windows.h>

const UCHAR N = 20, K = 9, Nn = 100;

struct elem

{

UINT quantity = 0;

char key[K]{ }, name[Nn]{ };

};

struct table

{

UCHAR n = 0;

elem \*elems[N]{ };

void insert(elem &e)

{

UCHAR p = binarySearch(e.key);

if (!strcmp(elems[p]->key, e.key))

elems[p] ->quantity += e.quantity;

else

{

shift(p);

elems[p] = &e;

}

}

int binarySearch(char \*key)

{

UCHAR l = 0, r = n;

for ( ; l < r; )

{

UCHAR m = (l + r) / 2;

if (strcmp(elems[m] ->key, key) < 0)

l = m + 1;

else

r = m;

}

return l;

}

void shift(UCHAR p)

{

n++;

for (CHAR i = n - 1; i >= p; i--)

elems[i + 1] = elems[i];

}

void deleteF(char \*key)

{

UCHAR p = binarySearch(key);

if (strcmp(key, elems[p] ->key)) p = 0;

if (p)

{

for (UCHAR i = p; i < n; i++)

elems[i] = elems[i + 1];

n--;

}

else printf\_s("Элемента в таблице нет.");

}

void input()

{

elem e;

FILE \*f = NULL;

fopen\_s(&f, "WORK.txt", "r");

for ( ; fscanf\_s(f, "%s", e.key, \_countof(e.key)) != EOF; isert(e))

fscanf\_s(f, "%s%u", e.name, \_countof(e.name), &e.quantity);

fclose(f);

}

void output()

{

FILE \*f = NULL;

fopen\_s(&f, "WORKOUT.txt", "w");

for (UCHAR i = 0; i < n; i++)

fprintf\_s(f, "%s %s %5u\n", elems[i]->key, elems[i] ->name, elems[i]->quantity);

fclose(f);

}

};

int main()

{

table \*t = new table;

UCHAR s = 0;

UINT outCp = GetConsoleOutputCP();

SetConsoleOutputCP(1251);

t->input();

scanf\_s("%hhu", &s);

t->deleteF(t->elems[s]->key);

t->output();

SetConsoleOutputCP(outCp);

return 0 \* \_getch();

}

**5 Набор тестов**

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Назначение** |
| **1** | 00000004 Lion 2112  00000002 Rela 257  00000001 Mami 25  00000002 Rela 4242  00000003 Limo 2552  00000004 Lion 2112  1 | Повторяющиеся ключи. Удаление второго элемента таблицы. |
| **2** | USARUS01 Guns 829421  0 | Таблицы из одного элемента, который и будет удален. |
| **3** | 00000007 Headlightу 32  00000001 Windshield 1  00000003 DoorMegali 4  00000009 HandleProp 128  00000004 RoofInkipe 8  00000005 TrunkTerro 146232  00000006 HoodErmolp 16  00000008 MirrorFeda 64  00000002 HonestUnik 2  4 | Все элементы таблицы различные. Удаление элемента посередине. |
| **4** | FJF53JF1 IphoneXpromax 421652  GFSK421H SamsungGalaxy 994241  LATWQG25 HuaweiMentali 0  FS638ORQ XiomiUndergro 9529529952  5 | Удаление несуществующего элемента таблицы. |

**6 Результаты работы программы**

|  |  |
| --- | --- |
| **№** | **Ввод и вывод программы** |
| **1** | 1  00000001 Mami 25  00000003 Limo 2552  00000004 Lion 4224 |
| **2** | 0 |
| **3** | 4  00000001 Windshield 1  00000002 HonestUnik 2  00000003 DoorMegali 4  00000004 RoofInkipe 8  00000006 HoodErmolp 16  00000007 Headlightу 32  00000008 MirrorFeda 64  00000009 HandleProp 128 |
| **4** | 5  Элемента в таблице нет.  FJF53JF1 IphoneXpromax 421652  FS638ORQ XiomiUndergro 9529529952  GFSK421H SamsungGalaxy 994241  LATWQG25 HuaweiMentali 0 |