**Сортировка массива методом Шелла**

Отчёт по практике

Выполнили: cт.гр. 97ЭЭ3 Толмач М., Ерегин П., Синева Т.

Пензенский государственный университет, Кафедра "Экономическая кибернетика"

Пенза 1998 г.

**Задание**

Цель работы: изучить метод Шелла для сортировки строкового массива и применить этот метод на практике.

Заданием на практическую работу является разработка программы на языке программирования Borland С++ v.3.1. для сортировки массива строк по их индексному значению. Значения элементов массива и их индексы задаются пользователем с клавиатуры.

**Введение**

В настоящее время индустрия производства компьютеров и программного обеспечения для них является одной из наиболее важных сфер экономики развитых стран. Ежегодно в мире продаются десятки миллионов компьютеров. Только в США объем продаж компьютеров составляет десятки миллионов долларов и постоянно продолжает расти.

В чем же причины такого стремительного роста индустрии персональных компьютеров и их сравнительная выгодность для многих деловых применений?

Простота использования, обеспеченная с помощью диалогового способа взаимодействия с компьютером.

Относительно высокие возможности по переработке информации, наличие программного обеспечения, а так же мощных систем для разработки нового программного обеспечения.

Язык С++ - универсальный язык общего назначения, область приложений которого - программирование систем в самом широком смысле. Кроме этого, С++ успешно используется как во многих приложениях, так и в мощных операционных системах.

**1 Входные данные**

Входными данными в программе является число элементов массива, значение индекса каждого элемента и строковые элементы массива. Все эти данные вводятся пользователем с клавиатуры по запросу программы. Число элементов массива не должно превышать 32767. Индексы элементов массива должны быть целочисленными значениями.

**2 Выходные данные**

Выходными данными в программе является исходный и отсортированный методом Шелла массив, которые выводятся на экран по запросу пользователя.

**3 Архитектура программы**

Данная программа разработана на языке программирования С++ и состоит из следующих функциональных модулей:

1) menu - обработчик меню;

2) input - ввод данных;

3) output - вывод данных;

4) sort - сортировка Шелла;

5) Основная программа.

1.Функция menu:

Осуществляет вывод на экран меню , опрос клавиатуры в бесконечном цикле и передвижение цветного курсора по пунктам меню. При нажатии клавиши ‘Esc’ возвращает -1, при нажатии клавиши с номером пункта меню возвращает этот номер, при нажатии клавиши ‘Enter’ возвращает номер текущего пункта меню.

Параметры для вызова функции menu: x,y – координаты меню на экране, \*capt – содержимое меню.

2.Функция input:

Осуществляет ввод индексов и элементов массива с клавиатуры, возвращает количество введенных элементов.

Параметры для вызова функции mas[] –указатель на массив, stn – номер первого вводимого элемента.

3.Функция output:

Осуществляет вывод содержимого массива на экран.

Параметры для вызова функции mas[] –указатель на массив, num – число элементов массива.

4.Функция sort:

Осуществляет сортировку массива по индексам элементов методом Шелла.

Сортировка методом Шелла заключается в следующем: сначала отдельно группируются и сортируются элементы, отстоящие друг от друга на расстоянии 9. После первого прохода элементы перегруппировываются и вновь сортируются элементы, отстоящие друг от друга на расстоянии 5, затем сортируются элементы,. отстоящие друг от друга на расстоянии 3, и наконец , на четвертом проходе идет обычная или одинарная сортировка.

Параметры для вызова функции mas[] –указатель на массив, num – число элементов массива.

5. Основная программа:

Осуществляет установку цветов, очистку экрана, вызов, функции обработки меню и в зависимости от возвращенного значения вызов одной из следующих функций: input, output, sort.

**Список литературы**

1. Вирт Н. Алгоритмы и структуры данных: Пер с англ. -М.: Мир,1989. - 360 с., ил.

2.Бьярн Страуструп. Язык программирования С++. в двух частях. Пер. с англ. Киев:"ДиаСофт",1993.-296 с. ил.

ПРИЛОЖЕНИЕ 1

Распечатка программы

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <string.h>

// Данные одного элемента массива

struct one\_elem {

int n; // Индекс

char st[100]; // Данные

};

// Обработка меню

int menu(int x,int y,char \* capt);

// Ввод данных

int input(one\_elem mas[],int stn);

// Вывод данных

void output(one\_elem mas[],int num);

// Сортировка Шелла

void sort(one\_elem mas[],int num);

// Обработка меню

int menu(int x,int y,char \* capt) {

int n,m; // Счетчики

int num; // Количество пунктов

int k; // Выбранный пункт

char \* pt; // Временный указатель на символ

char c; // Считанный с клавиатуры символ

// Вычисляем количество пунктов

num=strlen(capt)/20;

// Курсор на нулевой элемент

k=0;

// Бесконечный цикл обработки

for (;;) {

// Вывод меню

pt=capt;

for (n=0;n<num;n++) {

gotoxy(x,y+n);

// Закраска пункта, на который указывает курсор

if (n==k) {

// Закраска

textbackground(12);

textcolor(14);

} else {

// Нормальный

textbackground(3);

textcolor(1);

}

cprintf("%d) ",n+1);

for (m=0;m<20;m++) cprintf("%c",\*(pt++));

}

textbackground(3);

textcolor(1);

// Опрос клавиатуры

c=getch();

if (!c) c=getch();

// Проверка, не нажата ли клавиша с цифрой

if (((c-'1')>=0)&&((c-'1')<num)) {

// Установка указателя в зависимости от нажатой цифры

k=c-'1';

// Запись в буфер клавиатуры символа ENTER

ungetch(13);

} else {

// Анализ

switch(c) {

// Вверх

case (72):

if (k>0) k--; else k=num-1;

break;

// Вниз

case (80):

if (k<(num-1)) k++; else k=0;

break;

// Выход по ESC - возвращается -1

case (27):

return -1;

// Выход по ENTER - возвращается номер пункта

case (13): return k;

}

}

}

}

// Ввод данных

// Возвращаемое значение - количество введенных элементов

// Входные параметры - указатель на массив и номер первого вводимого элемента

int input(one\_elem mas[],int stn) {

clrscr(); // Очистка массива

int x; // Индекс

int num; // Количество введенных элементов

int n; // Счетчик

char a[100]; // Данные

// Ввод количества элементов

printf(" Число вводимых элементов: ");

scanf("%d",&num);

printf(" Вводите строчки формата X: Слово \n");

// Ввод элементов

for (n=0;n<num;n++) {

scanf("%d:%s",&x,a);

mas[n+stn].n=x;

strcpy(mas[n+stn].st,a);

};

return num;

}

// Вывод массива

// Входные параметры - указатель на массив и количество элементов

void output(one\_elem mas[],int num) {

clrscr();

int n; // Счетчик

printf(" Содержимое массива: \n");

printf(" Индекс Содержимое \n");

// Вывод элементов

for (n=0;n<num;n++)

printf("%5d %s\n",mas[n].n,mas[n].st);

// Ожидание ESC

gotoxy(1,24);

printf(" Нажмите ESC для продолжения ... ");

while (getch()!=27);

}

// Сортировка Шелла

void sort(one\_elem mas[],int num) {

int stp[4]={9,5,3,1}; // Шаги сортировки

int fs,mn,p; // Первый, минимальный и текущий элементы

int n; // Счетчик

one\_elem prm; // Временная переменная

// Цикл сортировки

for (n=0;n<4;n++) {

fs=0; // Начинать сортировать с начала

// Перебор всего массива

while (fs<num) {

// Поиск минимального элемента

p=fs;

mn=fs;

while (p<num) {

if (mas[p].n<mas[mn].n) mn=p;

p+=stp[n];

}

// Если минимальный элемент отличается от начального, поменять их местами

if (mn>fs) {

prm.n=mas[mn].n;

strcpy(prm.st,mas[mn].st);

mas[mn].n=mas[fs].n;

strcpy(mas[mn].st,mas[fs].st);

mas[fs].n=prm.n;

strcpy(mas[fs].st,prm.st);

}

fs+=stp[n]; // Переход к следующему элементу

}

}

}

// Основная программа

void main() {

one\_elem mas[100]; // Массив

int num; // Количество элементов

int st; // Выбранный пункт меню

textbackground(0);

textcolor(15);

clrscr();

do {

// Вывод меню

st=menu(30,5," Ввод данных "

" Добавление данных "

" Вывод данных "

" Сортировка Шелла "

" Выход из программы "

"\x0");

textbackground(0);

textcolor(15);

// Выполнение действий в зависимости от выбранного пункта

switch(st) {

// Ввод данных

case (0):

num=input(mas,0);

break;

// Добавление данных

case (1):

num+=input(mas,num);

break;

// Вывод массива

case (2):

output(mas,num);

break;

// Сортировка

case (3):

sort(mas,num);

break;

}

// Выход по ESC или последнему пункту

} while ((st<4)&&(st!=-1));

clrscr();

}

ПРИЛОЖЕНИЕ 2

Результаты работы программы

Меню:

1) Ввод данных

2) Добавление данных

3) Вывод данных

4) Сортировка Шелла

5) Выход из программы

1) Ввод данных:

Число вводимых элементов: 8

Вводите строчки формата X: Слово

0:sign

1001:else

3000:yes

1535:but

1:past

412:cell

99:alert

2888:object

3) Вывод данных:

Содержимое массива:

Индекс Содержимое

0 sign

1001 else

3000 yes

1535 but

1 past

412 cell

99 alert

2888 object

Нажмите ESC для продолжения ...

2) Добавление данных:

Число вводимых элементов: 1

Вводите строчки формата X: Слово

10000:hello

4) Сортировка Шелла

3) Вывод данных:

Содержимое массива:

Индекс Содержимое

0 sign

1 past

99 alert

412 cell

1001 else

1535 but

2888 object

3000 yes

10000 hello

Нажмите ESC для продолжения ...

5) Выход из программы

ПРИЛОЖЕНИЕ 3

Блок–схема алгоритма программы