Федеральное агенство связи

Федеральное государственное бюджетное образовательное учреждение высшего образования

"Сибирский государственный университет телекоммуникаций и информатики"

Лабораторная работа №6

по дисциплине "Структуры и алгоритмы обработки данных"

Выполнил: студент группы:

Проверил: ассистент кафедры ПМиК

Дьячкова И.С.

Новосибирск 2020

**Содержание:**

1. **Цель работы**
2. **Описание алгоритмов**
3. **Листинг программы**
4. **Результат работы программы**

**1.Цель работы:**

1). Разработать процедуру сортировки массива целых чисел методом Шелла (ShellSort).

2). Предусмотреть подсчет фактического количества пересылок и сравнений (Мф и Сф), сравнить с теоретическими оценками М и С.

5). Исследовать метод Шелла для n=100. 200, …500 на массивах случайных чисел и сравнить с методом прямого включения. Определить трудоемкость и количество К-сортировок по формуле Кнута. Построить таблицу вида:

|  |  |  |  |
| --- | --- | --- | --- |
| n | Количество  К-сортировок | Insert Мф+Сф | Shell Мф+Сф |
| 100 |  |  |  |
| 200 |  |  |  |
| 300 |  |  |  |
| 400 |  |  |  |
| 500 |  |  |  |

**2.Описание алгоритмов**

**void Fillinc(int A[N])**

Заполняет массив А возрастающими числами

**void FillDec(int A[N])**

Заполняет массив А убывающими числами.

**void FillRand(int A[N])**

Заполняет массив А случайными числами.

**void CheckSum(int A[N])**

Считает контрольную сумму элементов массива А.

**void RunNumber(int A[N])**

Считает серии в массиве А.

**void PrintMas(int A[N])**

Выводит на экран элементы массива А.

**void SelectSort (int A[], int n)**

Сортировки массива методом прямого выбора

**void BubbleSort (int A[], int n)**

Сортировки массива пузырьковым методом

**void ShakerSort(int A[], int n)**

Шейкерная сортировка

**void InsertSort (int A[], int n)**

Сортировка массива методом прямого включения

**3. Листинг программы**

#include <iostream>

#include <stdlib.h>

#include <stdio.h>

#include <graphics.h>

#include <time.h>

#include <math.h>

using namespace std;

int i=0, M = 0, C = 0, k=0, r=0,z;

int random (int N) {

return rand() % N;

}

void FillInc (int A[], int n) {

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

A[i] = i;

}

}

void FillDec (int A[], int n) {

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

A[i] = n-i;

}

}

void FillRand(int A[], int n) {

int i;

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

A[i] = 0;

}

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

A[i] = rand()% 10;

}

}

void PrintMas (int A[], int n) {

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

printf("%d ", A[i]);

}

}

void InsertSort (int A[], int n) {

int t=0, j=0;

C=0; M=0;

for (i=1; i<n; i++){

t=A[i];

j=i-1;

M++;

C++;

if ( j>-1 && t<A[j]) C--;

while (j>=0 && t<A[j]){

A[j+1]=A[j];

M++;

j=j-1;

C++;

}

A[j+1]=t;

M++;

}

}

int ShellSort (int A[], int n) {

int t=0, j=0;

C=0; M=0;

r=log2(n)-1;

z=r;

int \*h=new int[r];

h[0]=1;

for(i=1; i<r; i++){

h[i]=2\*h[i-1]+1;

printf(" h[%d]=%d ",i, h[i]);

}

for (k=h[r-1]; k>=1; k=h[r-1],r--){

for (i=k; i<=n; i++){

t=A[i];

j=i-k;

M++;

C++;

while (j>=0 && t<A[j]){

A[j+k]=A[j];

j=j-k;

M++;

C++;

}

A[j+k]=t;

M++;

}

}

}

int main () {

using namespace std;

int Sum,ser,n,y;

int w;

srand(time(NULL));

n=100;

int A[n];

printf("Убывающий массив: ");

FillDec(A, n);

PrintMas(A,n);

printf("\n ");

printf("\n ");

printf("Отсортированный массив: ");

ShellSort(A, n);

PrintMas(A,n);

printf("\n\n");

printf(" К-сортировки Shell Insert\n");

for(n=100;n<501;n+=100) {

int B[n], S[n];

C=0;M=0;

printf("n=%d",n);

FillRand(B,n);

ShellSort(B,n);

printf(" k=%d", z);

printf(" %d",M+C);

C=0;M=0;

FillRand(S,n);

InsertSort(S,n);

printf(" %d",M+C);

printf("\n\n");

C=0;M=0;

}

return 0;

}

**4. Результат работы программы**

