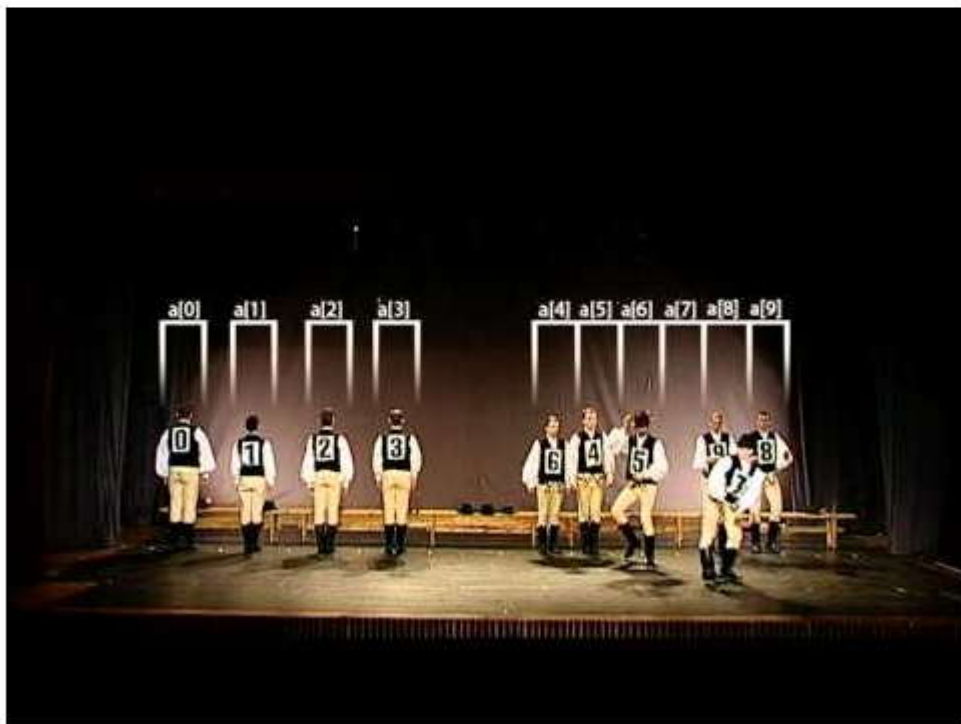


Чулюков В.А.

**МЕТОДЫ РАЗРАБОТКИ
ПРОГРАММ
(АЛГОРИТМЫ И
СТРУКТУРЫ ДАННЫХ)**

ЧАСТЬ 3

СОРТИРОВКИ



Воронеж - 2015

Лабораторная работа № 2

ТЕМА: Усовершенствованные методы сортировки массивов

ПРОДОЛЖИТЕЛЬНОСТЬ: 4 часа аудиторных занятий и 24 час. самостоятельной работы.

Задание 1. (Выполняется во время аудиторных занятий).

Наберите и отладьте программу, реализующую сортировку включениями с убывающими приращениями (сортировку Шелла):

```
program s6;
uses crt;
const n=10;
type item=record key:integer;
      end;
      index=-9..n;
var a:array[-9..n] of item;

procedure vvod_mass;
var i:index;
begin
randomize;
      for i:=1 to n do
            a[i].key:=random(100);
end; {of vvod_mass}

procedure ShellSort;
const t = 4;
var i,j,k,s:index; x:item; m:1..t;
h: array[1..t] of integer;
BEGIN
h[1]:=9; h[2]:=5; h[3]:=3; h[4]:=1;
      FOR m:=1 TO t DO
            BEGIN k:=h[m]; s:= -k;
                  FOR i:= k+1 TO n DO
                        BEGIN x:=a[i] ; j:=i-k;
                              IF s=0 THEN s:= -k; s:=s+1; a[s]:=x;
                              WHILE x.key<a[j].key DO
                                    BEGIN a [j +k]:=a[j]; j:=j-k
                                    END;
                              a[j +k]:=x
                        END
                  END
            END;

procedure vivod_mass;
var i:index;
begin
      for i:=1 to n do write(a[i].key:4);
```

```

    writeln;
end;
begin {of main}
clrscr;
vvod_mass;
vivod_mass;
writeln('nachat sortirovku? (enter)');
readln;
shellsort;
vivod_mass;
writeln('konec sort');
readln
end.

```

Вопросы для самопроверки:

1. Какая последовательность приращений используется в данной программе?

Самостоятельная работа:

1. Внесите в программу изменения, чтобы сортировались 10000 случайных чисел.
2. Выполните программу и засекуте время выполнения сортировки.
3. Убедившись, что засесть время выполнения сортировки с помощью часов не удастся – измените программу так, чтобы время сортировки вычислялось программой с помощью функции Milliseconds:

```

program s7;
uses dos, crt;
const n=10000;
type item=record key:integer;
    end;
    index=-9..n;
var a:array[-9..n] of item;

procedure vvod_mass;
var i:index;
begin
randomize;
    for i:=1 to n do
        a[i].key:=random(100);
end; {of vvod_mass}

procedure ShellSort;
const t = 4;
var i,j,k,s:index; x:item; m:1..t;
h: array[1..t] of integer;
BEGIN
h[1]:=9; h[2]:=5; h[3]:=3; h[4]:=1;
    FOR m:=1 TO t DO
        BEGIN k:=h[m]; s:= -k;
            FOR i:= k+1 TO n DO
                BEGIN x:=a[i] ; j:=i-k;

```

```

        IF s=0 THEN s:= -k; s:=s+1; a[s]:=x;
        WHILE x.key<a[j].key DO
        BEGIN a [j +k]:=a[j]; j:=j-k
        END;
        a[j +k]:=x
    END
END
Writeln(Milliseconds);
Readln
END;

procedure vivod_mass;
var i:index;
begin
    for i:=1 to n do write(a[i].key:4);
    writeln;
end;

begin {of main}
clrscr;
vvod_mass;
vivod_mass;
writeln('nachat sortirovku? (enter)');
readln;
shellsort;
writeln('konec sort');
readln
end.

```

Заметьте, что из основной программы удален вывод отсортированного массива для удобства наблюдения времени окончания сортировки.

Запишите длительность выполнения сортировки 10000 целых чисел.

4. Исследуйте длительность сортировки для 12000 чисел. Запишите результат.
5. Измените программу так, чтобы использовалась следующая последовательность приращений: 31, 15, 7, 3, 1. Отладьте и выполните ее для 12000 целых чисел. Запишите длительность выполнения сортировки.

Содержание отчета:

1. Текст исходной программы s2.
2. Текст программы для приращений 31, 15, 7, 3, 1 и с использованием процедуры GetTime.
3. Время выполнения сортировки для 10000 чисел с приращениями 9, 5, 3, 1.
4. Время выполнения сортировки для 12000 чисел с приращениями 9, 5, 3, 1.
5. Время выполнения сортировки для 12000 чисел с приращениями 31, 15, 7, 3, 1.
6. Краткие ответы на вопросы для самопроверки.

Задание 2. (Выполняется во время аудиторных занятий).

1. Наберите и отладьте программу, реализующую пирамидальную сортировку:

```
{R-}
program s8;
uses crt;
const n=10;
type item=record key:integer;
    end;
    index=1..n;
var a:array[1..n] of item;

procedure vvod_mass;
var i:index;
begin
randomize;
    for i:=1 to n do
        a[i].key:=random(100);
end; {of vvod_mass}

procedure HeapSort;
var l,r:index; x:item;

    procedure sift;
    label 13;
    var i,j: index;
    begin i:=l; j:=2*i; x:=a[i];
        while j<=r do
            begin if j<r then
                if a[j].key<a[j+1].key then j:=j+1;
                if x.key>=a[j].key then goto 13;
                a[i]:=a[j];
                i:=j;
                j:=2*i
            end;
            13: a[i]:=x
        end;
begin l:=(n div 2)+1; r:=n;
    while l>1 do
        begin l:=l-1; sift
        end;
    while r>1 do
        begin x:=a[l]; a[l]:=a[r]; a[r]:=x;
        r:=r-1; sift
        end
end;

procedure vivod_mass;
var i:index;
begin
    for i:=1 to n do write(a[i].key:4);
```

```

    writeln;
end;
begin {of main}
clrscr;
vvod_mass;
vivod_mass;
writeln('nachat sortirovku? (enter)');
readln;
Heapsort;
vivod_mass;
writeln('konec sort');
readln
end.

```

Вопросы для самопроверки:

1. Какие две функции выполняет процедура просеивания *sift*?

Самостоятельная работа:

1. Внесите в программу изменения, чтобы сортировались 10000 случайных чисел.
2. Измените программу так, чтобы время сортировки вычислялось программой с помощью функции *Milliseconds*.
3. Запишите время сортировки 10000 чисел.
4. Измерьте и запишите время выполнения сортировки 12000 чисел.

Содержание отчета:

1. Текст исходной программы s8.
2. Текст программы с использованием процедуры *GetTime*.
3. Время выполнения сортировки для 10000 чисел.
4. Время выполнения сортировки для 12000 чисел.
5. Выводы об эффективности пирамидальной сортировки по сравнению с сортировкой Шелла.
6. Краткие ответы на вопросы для самопроверки.

Задание 3. (Выполняется во время аудиторных занятий).

1. Наберите и отладьте программу, реализующую быструю сортировку.

```

{$R-}
uses dos,crt;
const n=16000;
type
    item=record key:integer;
    end;
    index=0..n;
var
    a:array[1..n] of item;
h, m, s, hund : Word;

```

```

procedure vvod_mass;
var i:index;
begin
randomize;
  for i:=1 to n do
    a[i].key:=random(100);
end; {of vvod_mass}

procedure QuickSort;
  procedure sort(l,r:index);
  var i,j:index; x,w:item;
  begin
    i:=l; j:=r;
    x:=a[(l+r) div 2];
    repeat
      while a[i].key<x.key do i:=i+1;
      while x.key<a[j].key do j:=j-1;
      if i<=j then
        begin
          w:=a[i];
          a[i]:=a[j];
          a[j]:=w;
          i:=i+1;
          j:=j-1
        end
      until i>j;
      if l<j then sort(l,j);
      if l<r then sort(i,r)
    end;
  begin
    sort(1,n)
  end;

procedure vivod_mass;
var i:index;
begin
  for i:=1 to n do write(a[i].key:4);
  writeln;
end;

begin {of main}
clrscr;
vvod_mass;
vivod_mass;
writeln('nachat sortirovku? (enter)');
readln;
QuickSort;
{vivod_mass;}
writeln('konec sort');
readln
end.

```

Вопросы для самопроверки:

1. Покажите часть кода, где выполняется разделение массива?
2. Какой элемент выбирается в качестве элемента x ?
3. Покажите часть кода, где выполняется рекурсия?

Самостоятельная работа:

1. Внесите в программу изменения, чтобы сортировались 10000 случайных чисел.
2. Измените программу так, чтобы время сортировки вычислялось программой с помощью функции `Milliseconds`.
3. Запишите время сортировки 10000 чисел.
4. Измерьте и запишите время выполнения сортировки 12000 чисел.
5. Составьте сравнительную таблицу:

№ п.п.	Название сортировки	Время сортировки массива из 10000 целых случайных чисел
...

6. Проанализируйте результаты, следующие из таблицы.

Содержание отчета:

1. Текст исходной программы.
2. Время выполнения сортировки для 10000 чисел.
3. Время выполнения сортировки для 100000 чисел.
4. Сравнительная таблица.
5. Анализ полученных результатов.