

Государственное бюджетное профессиональное образовательное учреждение
«Нижегородский радиотехнический колледж»

ОП.04 Основы алгоритмизации и программирования

ОТЧЁТ
по лабораторной работе № 4

Тема «Пузырьковая сортировка массива»

Выполнил:
обучающийся группы 2ИСиП-19-1
Мамонов Антон

Проверил:
Преподаватель
Калентьева Е.В.

Нижний Новгород
2020г.

Сформируйте случайным образом массив из 20 элементов. Значения элементов должны находиться в интервале от -50 до 50. 2. Создайте проект, выполняющий сортировку массива пузырьковым методом (предусмотрите вывод результата каждой итерации в сортировке).

```
program laba4;
const n=20; //кол-во элементов массива
type massiv=array [1..n] of integer; //задаем массив
var a:massiv; t,j,x,b,h:integer;
//a-массив; t-последний индекс замены; j-элемент массива;
//x-переменная для обмена; b-временная; h-вывод массива
begin
  randomize;
  for j:=1 to n do
    begin
      a[j]:=random(100)-50; //задаем элементам массива рандомные числа (-50;50)
      write(a[j], ' '); //выводим полученный массив
    end;
  writeln;
  writeln;
  b:=n; //задаем переменной индекс последнего элемента
  while b<>0 do //пока b не равно 0 - будет выполняться
    begin
      t:=0;
      for j:=1 to b-1 do //для всех элемент с 1 по b-1
        if a[j]>a[j+1] then
          //если значение элемента больше значения следующего
          begin
            x:=a[j]; //запоминаем элемент
            a[j]:=a[j+1]; //меняем элементы
            a[j+1]:=x; //и снова меняем
            t:=j; //запоминаем место последнего обмена
          end;
      b:=t; //присваиваем значение, если обмена не было - выход из цикла
      for h:=1 to n do //для всех элементов массива
        begin
          write(a[h], ' '); //выводим элемент при каждой итерации
        end;
      writeln;
    end;
  writeln;
  writeln;
  for j:=1 to n do write(a[j], ' '); //выводим готовый массив в строчку
end.
```

```

program laba4;
const n=20; //кол-во элементов массива
type massiv=array [1..n] of integer; //задаем массив
var a:massiv; t,j,x,b,h:integer;
//a-массив; t-последний индекс замены; j-элемент массива;
//x-переменная для обмена; b-временная; h-вывод массива
begin
  randomize;
  for j:=1 to n do
  begin
    a[j]:=random(100)-50; //задаем элементам массива случайные числа (-50;50)
    write(a[j], ' '); //выводим полученный массив
  end;
  writeln;
  writeln;
  b:=n; //задаем переменной индекс последнего элемента
  while b>0 do //пока b не равно 0 - будет выполняться
  begin
    t:=0;
    for j:=1 to b-1 do //для всех элементов с 1 по b-1
      if a[j]>a[j+1] then //если значение элемента больше значения следующего
      begin
        x:=a[j]; //запоминаем элемент
        a[j]:=a[j+1]; //меняем элементы
        a[j+1]:=x; //и снова меняем
        t:=j; //запоминаем место последнего обмена
      end;
    b:=t; //присваиваем значение, если обмена не было - выход из цикла
    for h:=1 to n do //для всех элементов массива
    begin
      write(a[h], ' '); //выводим элемент при каждой итерации
    end;
    writeln;
  end;
  for j:=1 to n do write(a[j], ' '); //выводим готовый массив в строку
end.

```

Окно вывода

-48 -47 -47 -38 -35 -33 -24 -16 -10 -4 -3 -3 1 3 15 20 27 35 35 38

Окно вывода | Список ошибок | Сообщения компилятора

Окно вывода

```

35 1 -47 -35 15 38 3 -47 -4 35 -3 20 -48 -3 -24 -10 -38 27 -16 -33
1 -47 -35 15 35 3 -47 -4 35 -3 20 -48 -3 -24 -10 -38 27 -16 -33 38
-47 -35 1 15 3 -47 -4 35 -3 20 -48 -3 -24 -10 -38 27 -16 -33 35 38
-47 -35 1 3 -47 -4 15 -3 20 -48 -3 -24 -10 -38 27 -16 -33 35 35 38
-47 -35 1 -47 -4 3 -3 15 -48 -3 -24 -10 -38 20 -16 -33 27 35 35 38
-47 -35 -47 -4 1 -3 3 -48 -3 -24 -10 -38 15 -16 -33 20 27 35 35 38
-47 -47 -35 -4 -3 1 -48 -3 -24 -10 -38 3 -16 -33 15 20 27 35 35 38
-47 -47 -35 -4 -3 -48 -3 -24 -10 -38 1 -16 -33 3 15 20 27 35 35 38
-47 -47 -35 -4 -48 -3 -24 -10 -38 -3 -16 -33 1 3 15 20 27 35 35 38
-47 -47 -35 -48 -4 -24 -10 -38 -3 -16 -33 -3 1 3 15 20 27 35 35 38
-47 -47 -48 -35 -24 -10 -38 -4 -16 -33 -3 -3 1 3 15 20 27 35 35 38
-47 -48 -47 -35 -24 -38 -10 -16 -33 -4 -3 -3 1 3 15 20 27 35 35 38
-48 -47 -47 -35 -38 -24 -16 -33 -10 -4 -3 -3 1 3 15 20 27 35 35 38
-48 -47 -47 -38 -35 -24 -33 -16 -10 -4 -3 -3 1 3 15 20 27 35 35 38
-48 -47 -47 -38 -35 -33 -24 -16 -10 -4 -3 -3 1 3 15 20 27 35 35 38
-48 -47 -47 -38 -35 -33 -24 -16 -10 -4 -3 -3 1 3 15 20 27 35 35 38
-48 -47 -47 -38 -35 -33 -24 -16 -10 -4 -3 -3 1 3 15 20 27 35 35 38

```

Окно вывода | Список ошибок | Сообщения компилятора

Компиляция прошла успешно (28 строк) 4 предупреждений