Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

|  |
| --- |
|  |
|  |

# оТЧЕТ

по лабораторной работе

на тему:

Рекурсия

|  |  |  |
| --- | --- | --- |
| Выполнил  Студент гр. 151001 |  | В. В. Лозюк |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2022

1. Теоретические сведения по теме лабораторной работы

**Алгоритм** – система правил, четко описывающая последовательность действий, которые необходимо выполнить для решения задачи.

**Свойства правильного алгоритма**:

1. ***Дискретность*** – значение величин в каждый следующий момент времени должны получаться по определенным правилам из значений величин, имевшихся в предшествующий момент времени.
2. ***Определенность (детерминированность)*** – каждое правило алгоритма должно быть однозначным. Значения величин, получаемые в какой-то момент времени, однозначно связаны со значениями величин, вычисленных ранее.
3. ***Результативность (конечность)*** – алгоритм должен приводить к решению задачи за конечное число шагов.
4. ***Массовость*** – алгоритм должен разрабатываться в общем виде так, чтобы его можно было применить для класса задач, различающихся лишь исходными данными.

**Способы описания алгоритма:**

1. Запись на естественном языке (словесное описание)
2. Изображение в виде схемы (графическое описание)
3. Запись на алгоритмическом языке (составление программы)

**Типы алгоритмов:**

1. линейные
2. разветвляющиеся
3. циклические

**Цикл** – участок схемы, многократно повторяемый в ходе вычислений.

**Классификация циклов:**

1. По взаимному расположению:
   1. Простые;
   2. Сложные - называется цикл, если он содержит в себе другой, вложенный в него цикл;

c. Внутренние и Внешние (В теле любого оператора **цикла** могут находиться другие операторы **цикла**. При этом **цикл**, содержащий в себе другой, называют **внешним**, а **цикл**, находящийся в теле первого — **внутренним** (вложенным)).

1. По местоположению условия выполнения цикла:

**С предусловием** — цикл, который выполняется, пока истинно некоторое условие, указанное перед его началом.

**С постусловием** — цикл, в котором условие проверяется *после* выполнения тела цикла.

1. По виду условия выполнения:

а)**С параметром** — цикл, в котором переменная изменяет свое значение от начального значения до конечного с шагом, определенным заранее, и для каждого из значений переменной выполняется цикл.

б)**Итерационные** — цикл, для которого число повторений тела цикла заранее неизвестно.

1. Задание на лабораторную работу

Отображение всех перестановок элементов в массиве.

1. Выполнение
   1. Разработка алгоритма

Таблица 3.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя идентификатора | Назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| Arr | Массив для генерирования перестановок | array[1..N] of integer; | - | - | - |

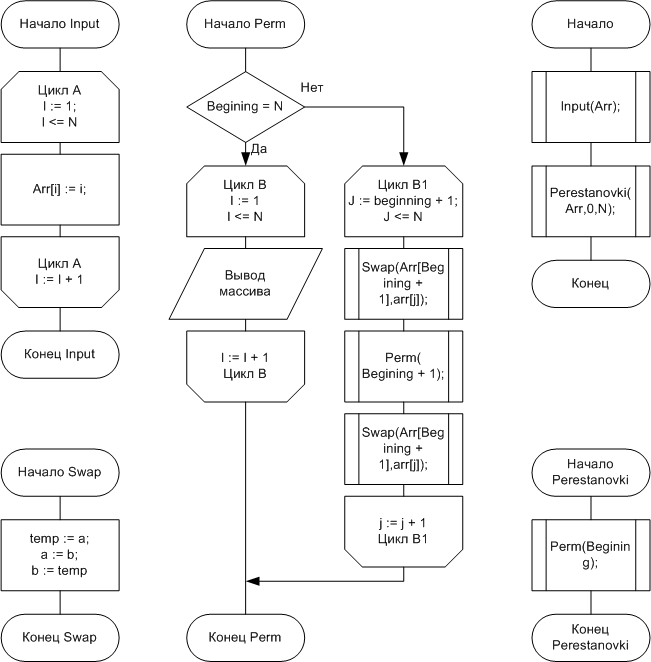


Рисунок 3.1 – Схема работы программы

Текст программы и его описание

program Perm;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

Const

N = 4;

Type

Tarr = array[1..N] of integer;

procedure Input(var Arr:Tarr);

var

i: integer;

begin

for i := 1 to N do

Arr[i] := i;

end;

procedure Swap(var a,b:integer);

var

temp : integer;

begin

temp := a;

a := b;

b := temp;

end;

procedure Perestanovki(Arr:TArr; Begining, N:integer);

var i,j:integer;

begin

if Begining = N then

begin

for i := 1 to N do

write(Arr[i]);

writeln;

end

else

for j := Begining + 1 to N do

begin

Swap(Arr[Begining + 1],arr[j]);

perestanovki(Arr, Begining + 1, n);

Swap(Arr[Begining + 1],arr[j]);

end;

end;

var

Arr: Tarr;

begin

Writeln('Перестановки среди ',n,' элементов:');

Input(Arr);

Perestanovki(Arr,0,N);

readln;

end.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
| Проверка на работоспособность | 1 | 3 |  | Тест пройден |

Тесты пройдены успешно.

* 1. Итоговый текст программы

program Perm;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

Const

N = 3;

Type

Tarr = array[1..N] of integer;

procedure Input(var Arr:Tarr);

var

i: integer;

begin

for i := 1 to N do

Arr[i] := i;

end;

procedure Swap(var a,b:integer);

var

temp : integer;

begin

temp := a;

a := b;

b := temp;

end;

procedure Perestanovki(Arr:TArr; Begining, N:integer);

var i,j:integer;

begin

if Begining = N then

begin

for i := 1 to N do

write(Arr[i]);

writeln;

end

else

for j := Begining + 1 to N do

begin

Swap(Arr[Begining + 1],arr[j]);

perestanovki(Arr, Begining + 1, n);

Swap(Arr[Begining + 1],arr[j]);

end;

end;

var

Arr: Tarr;

begin

Writeln('Перестановки среди ',n,' элементов:');

Input(Arr);

Perestanovki(Arr,0,N);

readln;

end.