

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №4

по дисциплине: Основы программирования

тема: «Обработка одномерных массивов с использованием подпрограмм»

Выполнил: ст. группы ПВ202

Аладиб язан

Проверил:

Валентина Станиславовна

Лабораторная работа №4

«Обработка одномерных массивов с использованием подпрограмм»

Цель работы: получение навыков работы с массивами и подпрограммами

Задания для подготовки к работе :

1. Изучите способы описания и использования массивов, базовые алгоритмы обработки массивов.
2. Изучите виды и назначение подпрограмм, правила их описания и вызова.
3. Разбейте задачу соответствующего варианта на подзадачи, таким образом, чтобы решение каждой подзадачи описывалось подпрограммой, а основная программа состояла бы в основном из вызовов подпрограмм.
4. Опишите блок-схему алгоритма решения задачи в укрупненных блоках.
5. Опишите используемые структуры данных, спецификации и блок-схемы подпрограмм, соответствующих укрупненным блокам. Спецификация содержит: заголовок подпрограммы, назначение, входные и выходные параметры.
6. Опишите блок-схему алгоритма решения задачи с использованием блоков «предопределенный процесс».
7. Закодируйте алгоритм.
8. Подберите наборы тестовых данных с обоснованием их выбора.

Задания к работе:

1. Наберите программу, отладьте ее и протестируйте.
2. Выполните анализ ошибок, выявленных при отладке программы.

Задания варианта №2:

Если данная последовательность не упорядочена ни по неубыванию, ни по невозрастанию, найти среднее геометрическое положительных членов.

Выполнение работы:

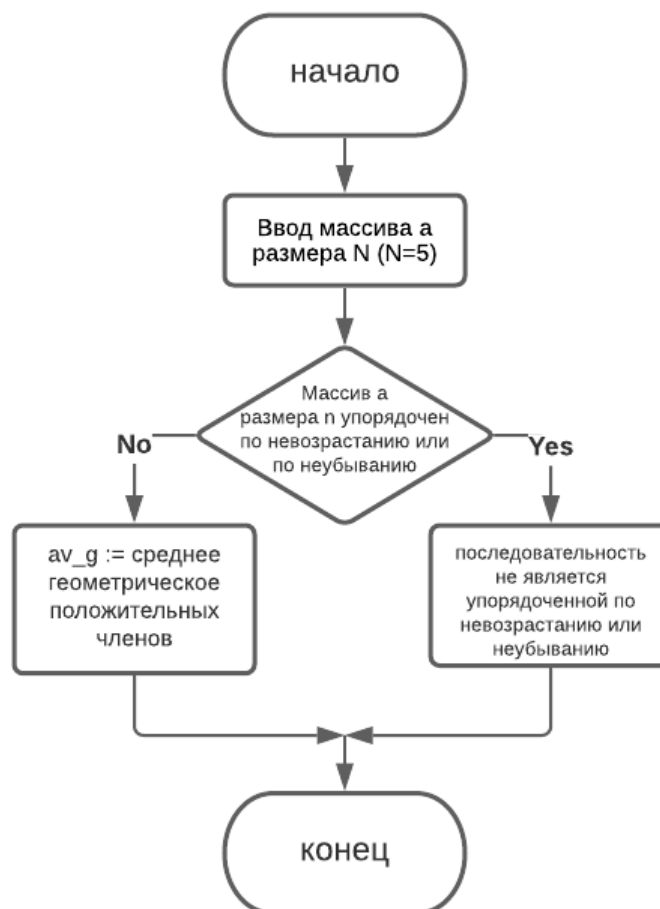
Выделение подзадач

Выделим следующие подзадачи:

- Ввод массива a размера N ($N=5$)
- Проверка последовательность не является упорядоченной по невозрастанию или неубыванию
- Вычисление и вывод среднее геометрическое положительных членов

Опишем алгоритм в укрупненных блоках в терминах выделенных подзадач

Блок-схема алгоритма в укрупненных блоках:



Описание структур данных:

```
const  
    N = 5;  
type t_range = 1..N;  
t_arr = array [t_range] of integer;
```

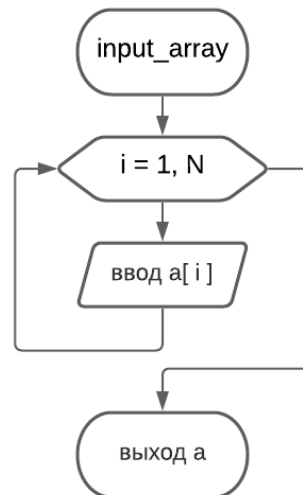
Описание подпрограмм

процедура (input_array):

Спецификация:

1. Заголовок: input_array(var a: t_arr)
2. Назначение: Ввод массива a размера N (N=5)
3. Входные параметры: нет
4. Выходные параметры: a

Блок-схема:

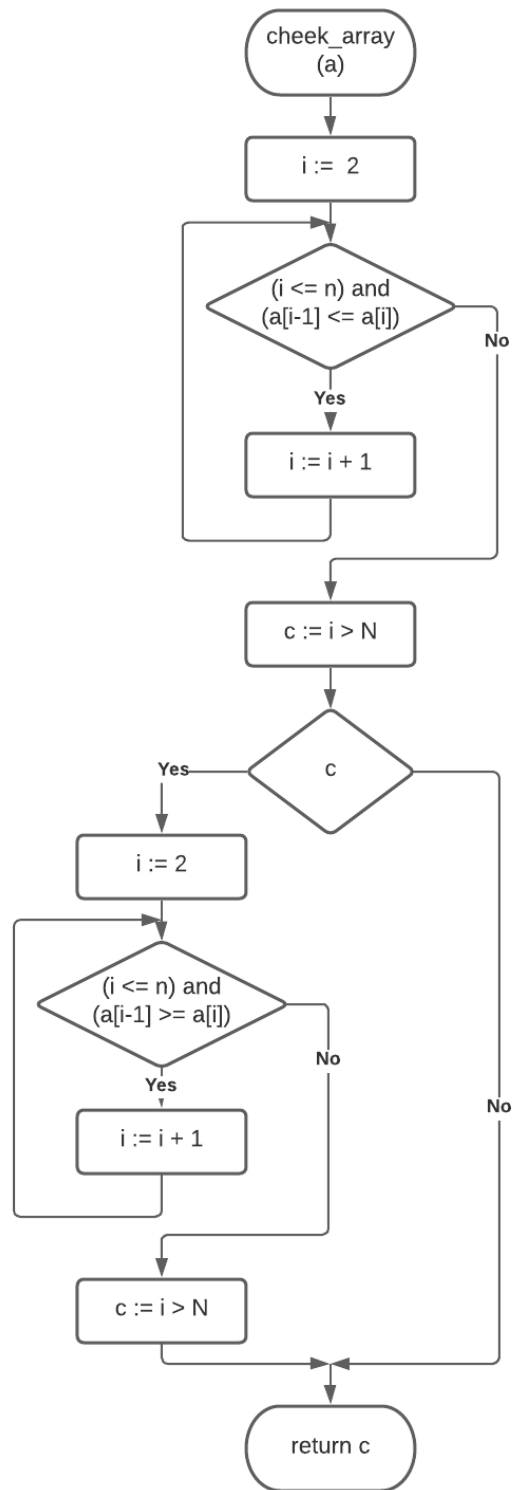


функция (check_array):

Спецификация :

1. Заголовок: function check_array(a: t_arr)
2. Назначение: Возвращает значение истина если Массив a размера N упорядочен по невозрастанию или по неубыванию , иначе - "ложь".
3. Входные параметры: a
4. Выходные параметры: нет

Блок-схема:

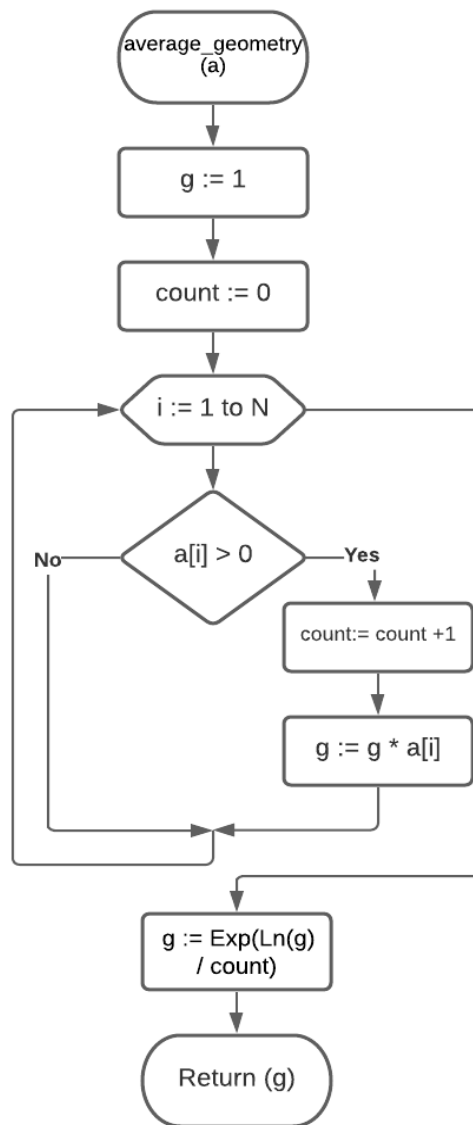


функция (average_geometry):

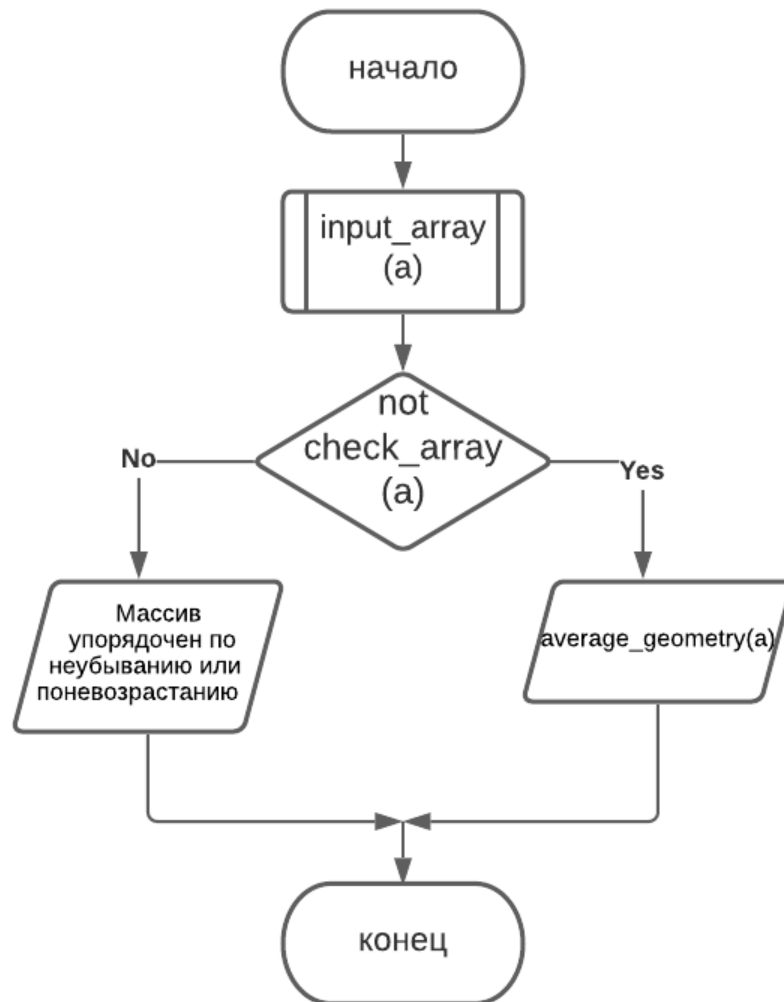
Спецификация :

1. Заголовок: function average_geometry(a: t_arr): real;
2. Назначение: возвращает среднее геометрическое положительных членов
3. Входные параметры: a
4. Выходные параметры: нет

Блок-схема:



Блок-схема алгоритма решения задачи с блоками «предопределенный процесс»:



Тестовые данные:

| Исходные данные | Результаты |
|---------------------|--|
| 23 34 11 22 68 | Среднее геометрическое положительных членов = 26.41 |
| 1 2 3 4 5 | Массив упорядочен по неубыванию или по невозрастанию |
| -12 -35 -91 -11 -24 | Среднее геометрическое положительных членов = NaN |
| 11 22 33 44 55 | Массив упорядочен по неубыванию или по невозрастанию |

Текст программы:

```
program lab4;
const
  N = 5;
type t_range = 1..N;
t_arr = array [t_range] of integer;

procedure input_array(var a: t_arr);
var
  i: integer;

begin
  for i := 1 to N do
    read(a[i])
  end;

function check_array(a: t_arr): boolean;
var
  i: integer;
  c: boolean;
begin
  i := 2;
  while (i <= N) and (a[i - 1] <= a[i]) do
    i := i + 1;
  c := i > N;
  if c then
    check_array := c
  else
    begin
      i := 2;
      while (i <= N) and (a[i - 1] >= a[i]) do
        i := i + 1;
      c := i > N;
      if c then
        check_array := c;
      end;
    end;
  end;

function average_geometry(a: t_arr): real;
var
  i, count: integer;
  g: real;
begin
  g := 1;
```



```

count := 0;
for i := 1 to N do
begin
    if a[i] > 0 then
    begin
        count := count + 1;
        g := g * a[i];
    end;
end;
g := Exp(Ln(g) / count);
average_geometry := g;
end;

var
a: t_arr;
g: real;

begin
writeln('введите массива');
input_array(a);
if not check_array(a) then
    write('Среднее геометрическое положительных членов =', average_geometry(a))
else
    write('Массив упорядочен по неубыванию или по невозрастанию')
end.

```

The screenshot shows the PascalABC.NET 3.7.1 IDE. The main window displays the source code of a Pascal program. Below the code editor is the Output Window, which shows the program's execution results. The status bar at the bottom indicates that the compilation was successful with 68 lines of code and 3 warnings.

Source Code (Program1.pas):

```

if a[i] > 0 then
begin
    count := count + 1;
    g := g * a[i];
end;
end;
g := Exp(Ln(g) / count);
average_geometry := g;
end;

var
a: t_arr;
g: real;

begin
writeln('введите массива');
input_array(a);
if not check_array(a) then
    write('Среднее геометрическое положительных членов =', average_geometry(a))
else
    write('Массив упорядочен по неубыванию или по невозрастанию')
end.

```

Output Window:

```

введите массива
23
34
11
22
68
Среднее геометрическое положительных членов =26.4183814357166

```

Status Bar: Compilation is completed successfully (68 lines), 3 warnings. Ln 68 Col 5 33%

