**Условие задачи**

Разработать алгоритм и программу с единственным циклом для обработки одномерных массивов.



**Первый шаг в решении задачи – анализ данных.**

Прежде всего необходимо учесть, чтобы наш алгоритм сооветсвовал нескольким важным факторам:

* Использование алгоритма было удобным;
* Алгоритм был простым и наглядным;
* Алгоритм самого решения должен быть однопроходным;

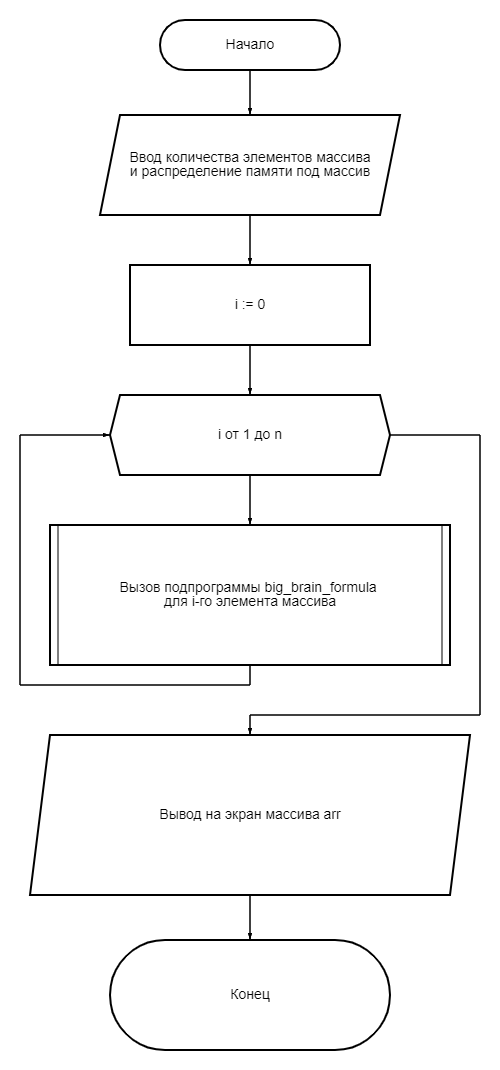
Для данной задачи необходимо задавать количество элементов массива и значения иниализирующие элементы этого массива.

**Метод решения** задачи состоит в вычислении результата функции f(x) =   
для заданного множества X.

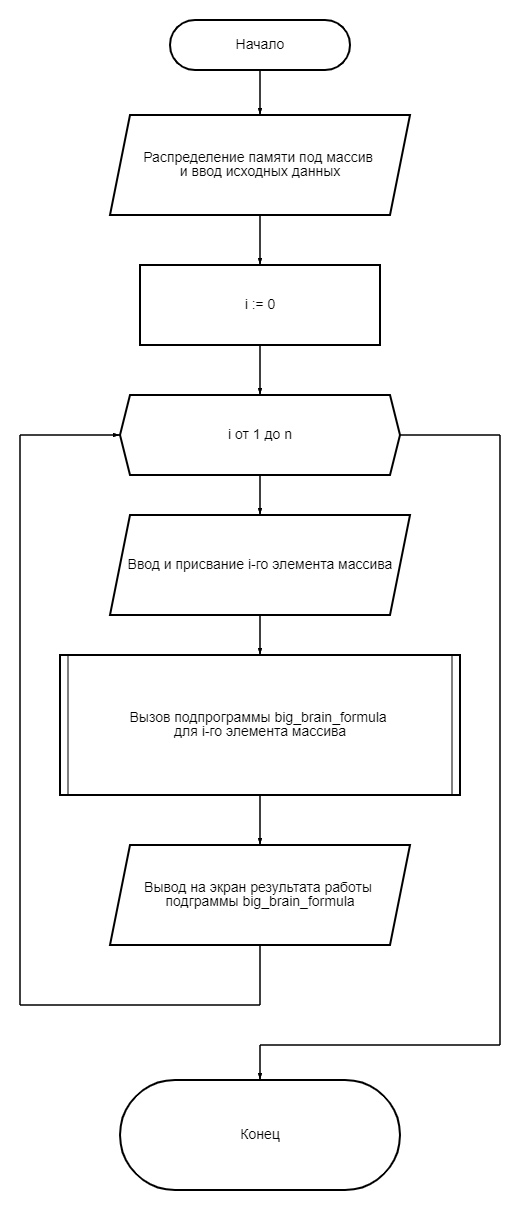
**Состав данных**

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Смысл** | **Тип** | **Структура** |
| **Исходные данные** | | | |
| *n* | количество элементов массива | целый | переменая |
| *arr* | множество значений x | массив вещественных чисел | одномерный динамический массив вещественного типа |
| **Выходные данные** | | | |
| *y* | результата функции f(x) | вещественный | переменная |
| **Промежуточные данные** | | | |
| *i* | переменная цикла | целый | переменная |

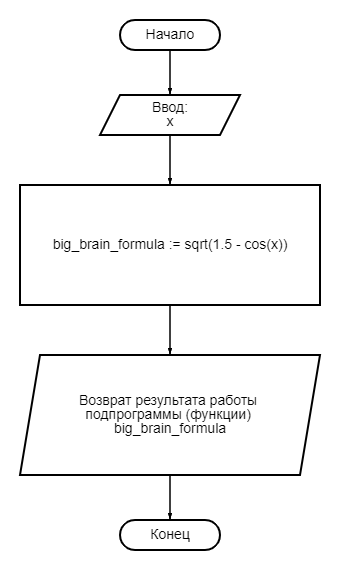
**Блок-схема алгоритма (Вариант 1)**

****

**Блок-схема алгоритма (Вариант 2)**



**Блок подпрограммы big\_brain\_formula**



**Форма ввода**

n

<*n*>

arr[0]=*<* arr[0]>

…

<*arr[i]*> <y[i]>

**Форма вывода**

x y

<*arr[0]*> <y[0]>

…

<*arr[i]*> <y[i]>

**Листинг программы написанной на языке object pascal и скомпилированной с помощью freepascal**

**program** lab\_1\_4\_1;

**uses** Math, sysutils;

**function** big\_brain\_formula(x: real): real;

**begin**

big\_brain\_formula := sqrt(1.5 - cos(x));

**end**;

**var**

y, array\_element: real;

n, i: integer;

arr: **array of** real;

**begin**

writeln('n');

readln(n);

setlength(arr, n);

**for** i := 0 **to** n - 1 **do**

**begin**

write('arr[', i, ']=');

read(arr[i]);

**end**;

writeln('x y');

**for** array\_element **in** arr **do**

**begin**

writeln(array\_element:2:2, ' ', big\_brain\_formula(array\_element):2:2);

**end**;

**end**.