**Лабораторная работа №10**

**Итерационные ЦВП с управлением по индексу и функции. Вариационный ряд.**

2. Цель: научиться создавать и применять Итерационные циклические вычислительные процессы с управлением по индексу и функции.

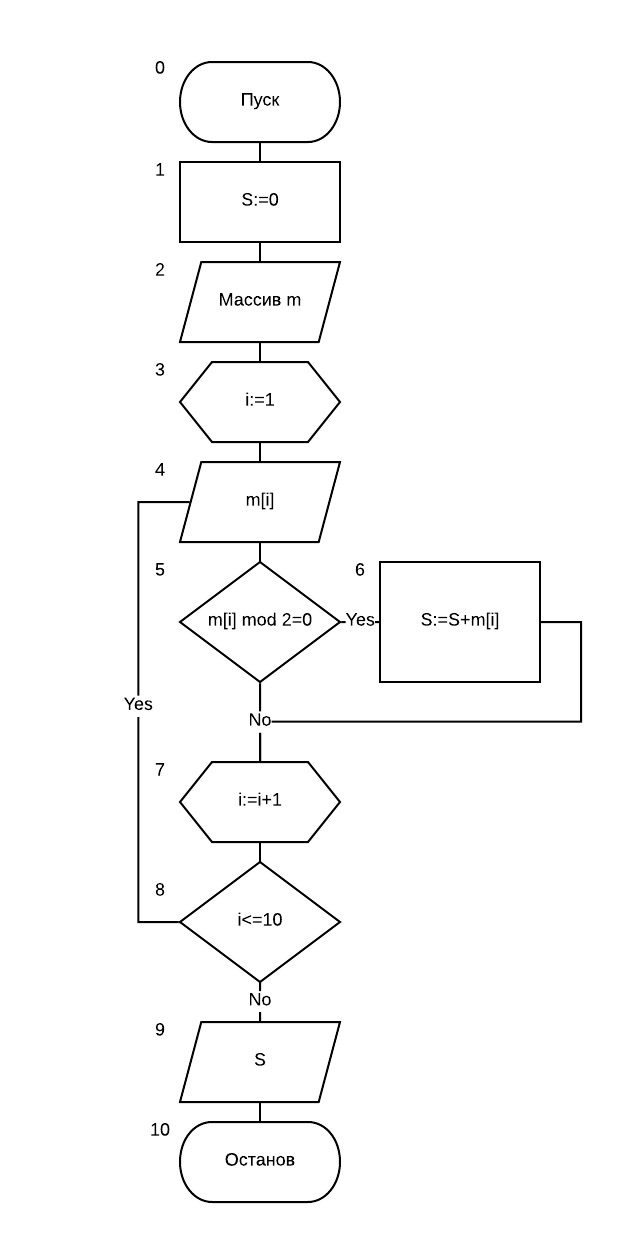
3. Оборудование: ПК, Lazarus, PascalABC.

**Задание №1**

4. Дан одномерный массив. Найти сумму четных (по значению) элементов массива.

5. m[i] mod 2

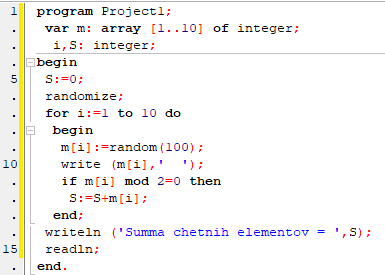
6.



7.

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| i | Индекс массива | integer |
| S | Сумма | integer |
| m | массив | Array of integer |

8.



9. 

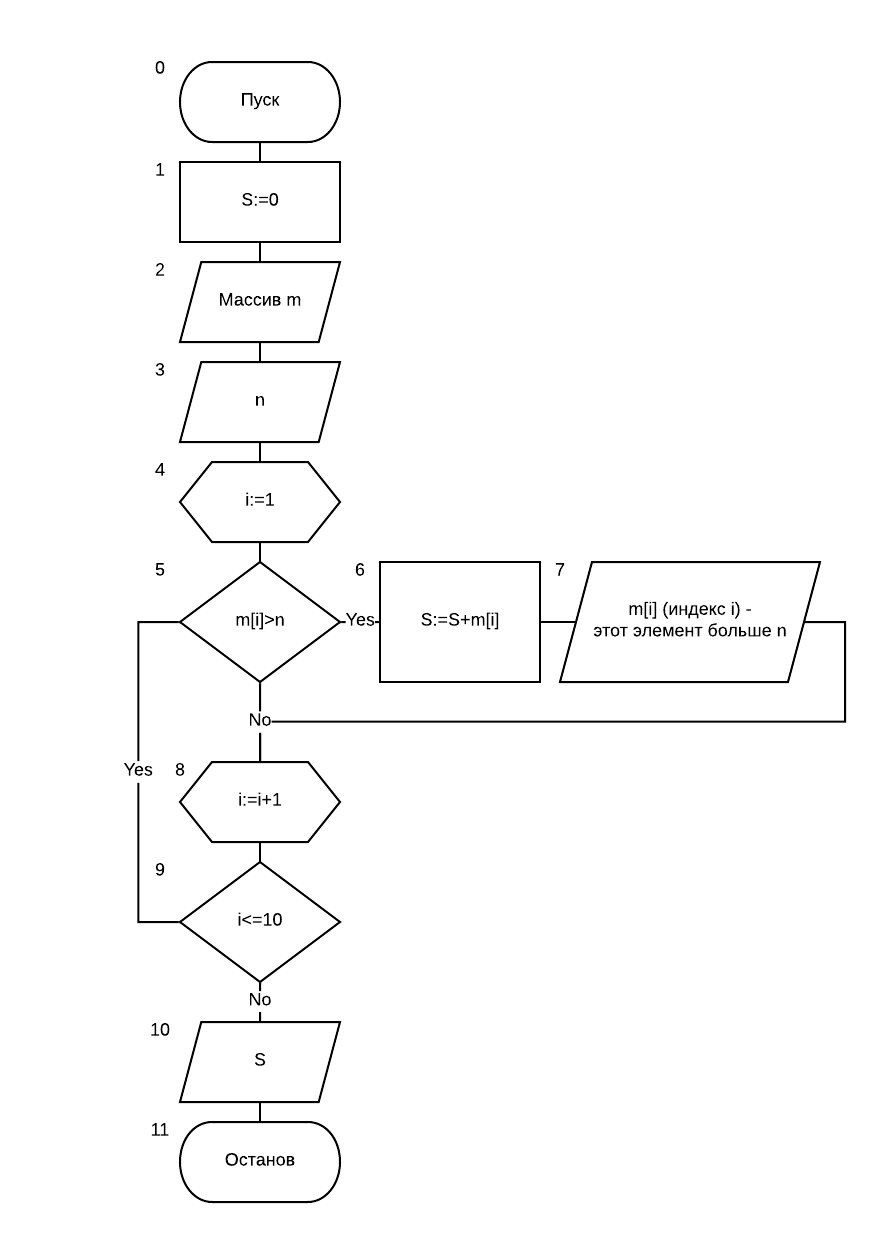
10. Чтобы определить, какие элементы массива являются чётными, программа определят остаток после целочисленного деления на 2. Если остаток равен нулю, то этот элемент добавляется в сумму.

**Задание №2**

4. Дан одномерный массив. С клавиатуры вводится число. Найти сумму всех элементов массива, значение которых больше числа, введенного с клавиатуры и вывести их индексы.

5. m[i]>n S:=S+m[i]

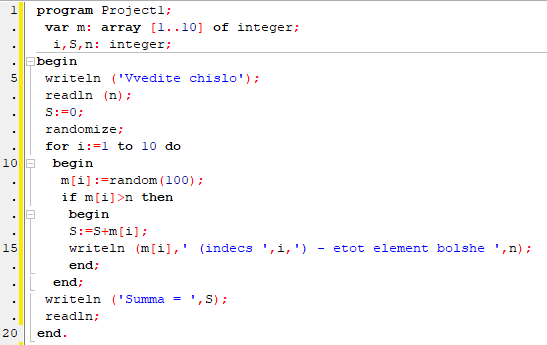
6.

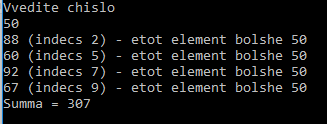


7.

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| i | Индекс массива | integer |
| S | Сумма | integer |
| n | Вводимое число | integer |
| m | массив | Array of integer |

8.



9. 

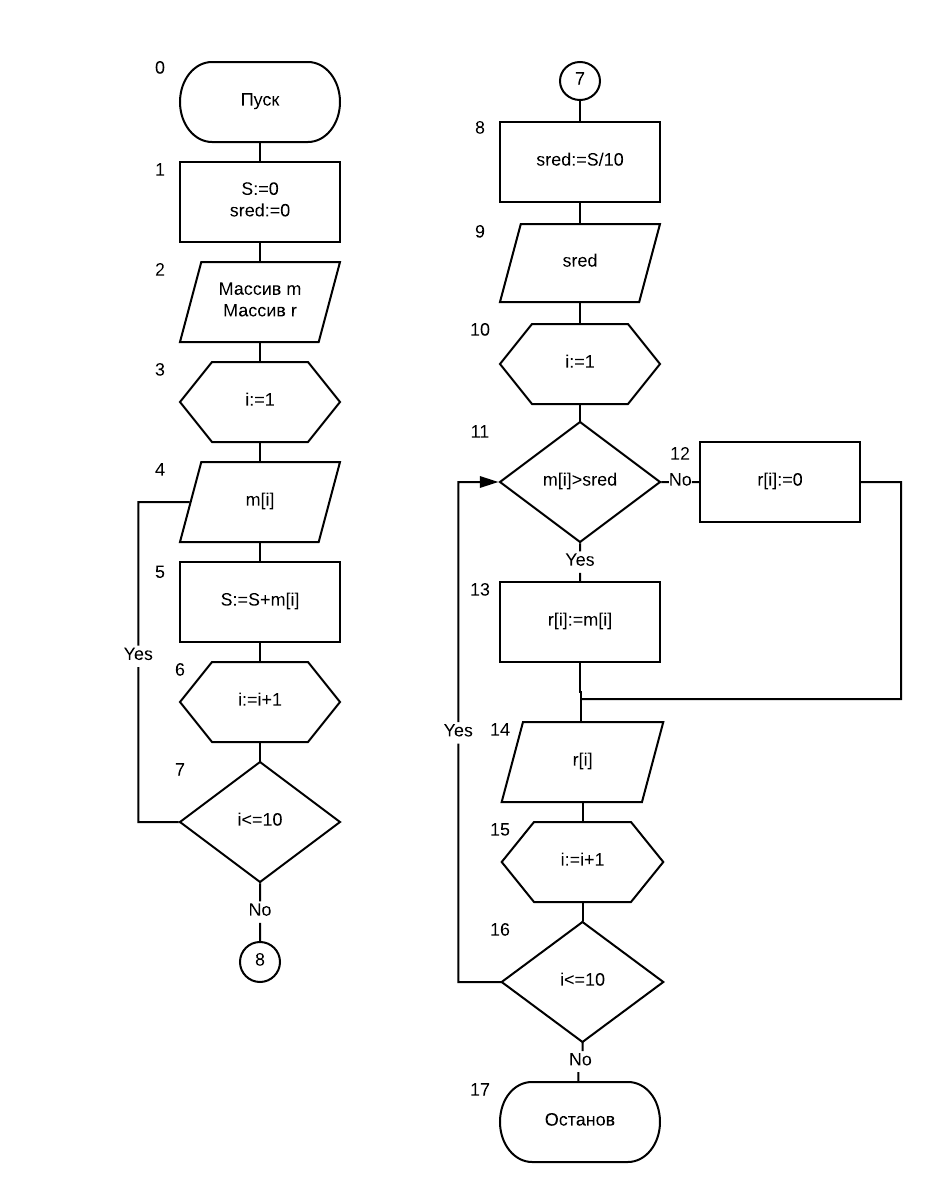
10. На входе в цикл проверяется, является ли этот элемент массива больше числа, введённого пользователем. Если является, то этот элемент прибавляется к накопителю суммы и выводится на экран вместе со своим индексом.

**Задание №3**

4. Дан одномерный массив. Найти его среднее арифметическое. Составить второй массив, элементами которого будут элементы первого массива, которые больше среднего арифметического. Остальные элементы заменить нулями.

5. S:=S+m[i] sred:=S/10

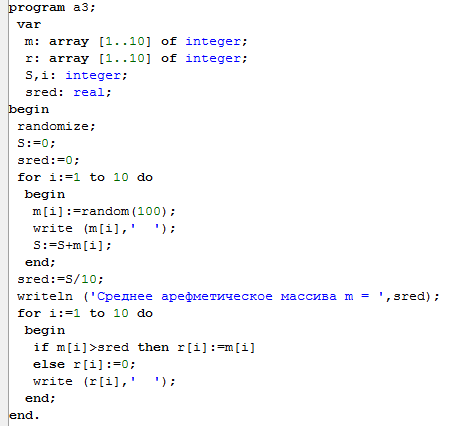
6.

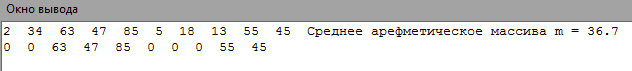


7.

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| m | начальный массив | Array of integer |
| r | Получившийся массив | Array of integer |
| S | Накопитель суммы | Integer |
| i | Индекс массивов | Integer |
| sred | Среднее арифметическое массива m | Real |

8.



9. 

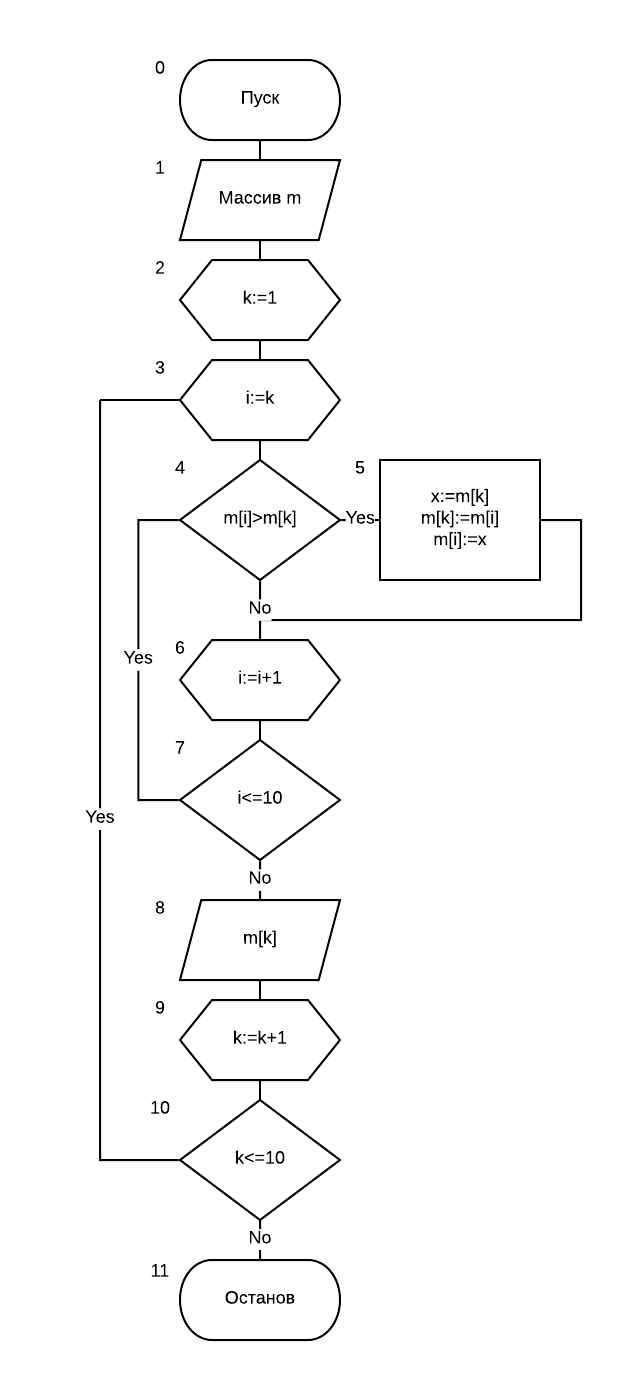
10. Первый цикл направлен на суммирование всех элементов массива m. После массива в переменную sred записывается среднее арифметическое массива m. После этого во втором цикле элементу массива r присваивается то же значение, что и у элемента массива m с тем же индексом, если этот элемент массива m больше среднего арифметического массива m. Если же этот элемент меньше, то элементу массива r присваивается значение 0.

**Задание №4**

4. Дан одномерный массив. Упорядочить его по убыванию.

5. x:=m[k] m[k]:=m[i] m[i]:=x

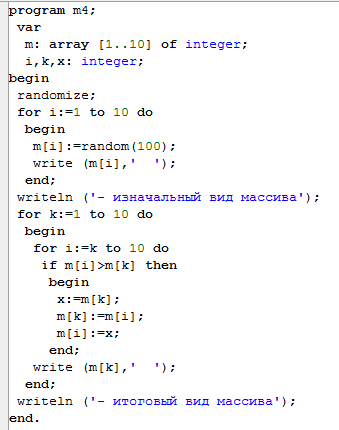
6.

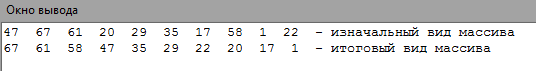


7.

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| m | массив | Array of integer |
| i | Индекс/параметр внутреннего цикла | integer |
| k | Индекс/параметр внешнего цикла | integer |
| x | Промежуточная переменная (запоминает значение m[k]) | integer |

8.



9. 

10. Для упорядочивания элементов массива по убыванию программа использует массив в массиве. Внутренний массив перебирает все элементы от к-того до последнего. Если какой-то элемент больше к-того, то алгоритм меняет их местами. Таким образом наибольшее число окажется правее. Внешний цикл перебирает значения к от 1 до 10 и после выполнения внутреннего цикла выводит к-ый элемент.

11. Вывод: Итерационные циклические вычислительные процессы с управлением по индексу и функции отлично подходят для различных работ с элементами массива: будь то нахождение суммы определённых элементов или же сортировка по определённому признаку.