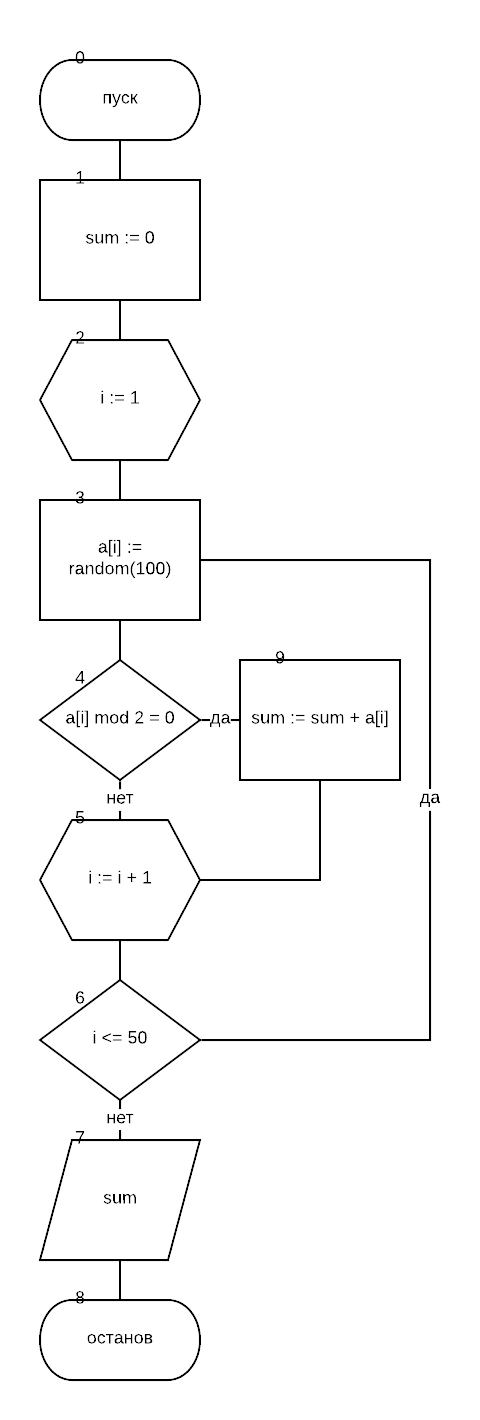
# Лабораторная работа № 10

## 1 задание

1. Разветвляющиеся вычислительные процессы. Оператор выбора.
2. Научится реализовать алгоритм для вычисление разветвляющихся вычислительных процессов средствами компилятора и языка программирования Pascal.
3. ПК, PascalABC.NET 3.4.2.
4. Дан одномерный массив. Найти сумму четных (по значению) элементов массива.
5. sum := sum + a[i].
6. 

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| i | Счётчик цикла | integer |
| sum | Сумма | integer |

**program** pr10;

**var**

a: **array** [1..100] **of** integer;

i, sum: integer;

**begin**

randomize;

sum := 0;

**for** i:=1 **to** 50 **do**

**begin**

a[i] := random(100);

**if** a[i] **mod** 2 = 0 **then**

**begin**

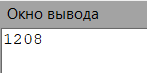
sum := sum + a[i]

**end**

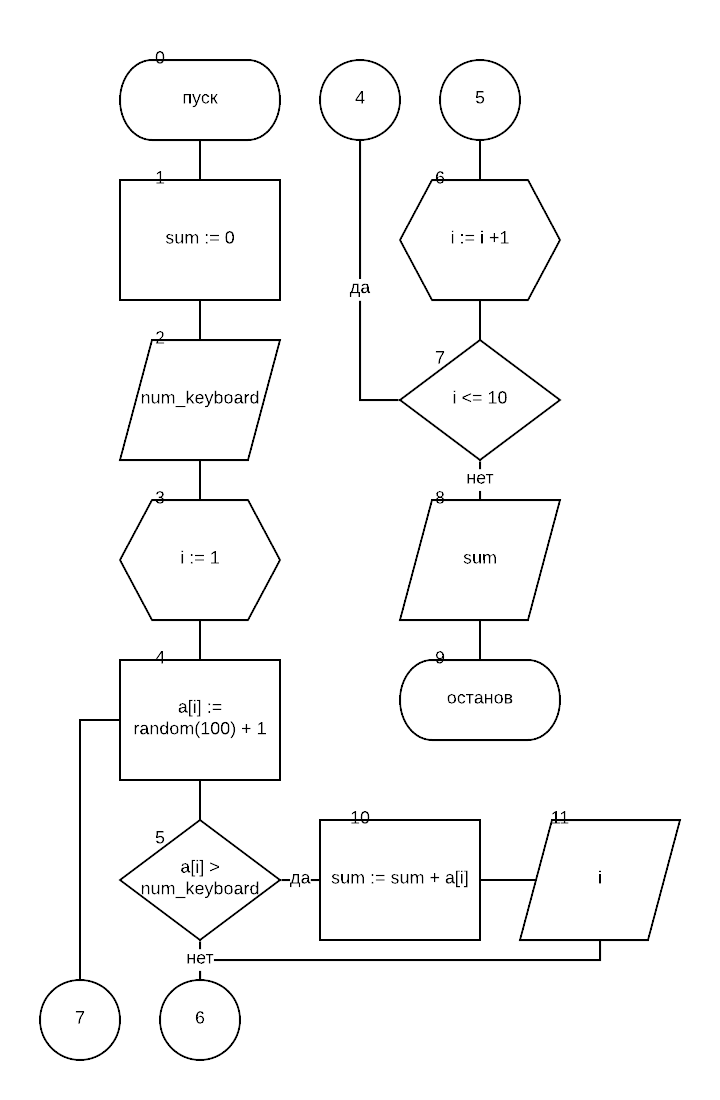
**end**;

writeln(sum)

**end**.

1. 
2. Использовал цикл for для заполнения массива.
3. Нашёл сумму четных (по значению) элементов данного массива.

## 2 задание

1. Разветвляющиеся вычислительные процессы. Оператор выбора.
2. Научится реализовать алгоритм для вычисление разветвляющихся вычислительных процессов средствами компилятора и языка программирования Pascal.
3. ПК, PascalABC.NET 3.4.2.
4. Дан одномерный массив. С клавиатуры вводится число. Найти сумму всех элементов массива, значения которых больше числа, введенного с клавиатуры и вывести их индексы.
5. sum := sum + a[i].
6. 

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| i | Счётчик цикла | integer |
| sum | Сумма | integer |
| num\_keyboard | Число, кот. юзер вводит с клаиватуры | integer |

**program** pr10;

**var**

a: **array** [1..10] **of** integer;

i, sum, num\_keyboard: integer;

**begin**

sum := 0;

write('Введите число: ');

read(num\_keyboard);

write('Индексы: ');

**for** i := 1 **to** 10 **do begin**

randomize;

a[i] := random(100) + 1;

**if** a[i] > num\_keyboard **then begin**

sum := sum + a[i];

write(i, ' ')

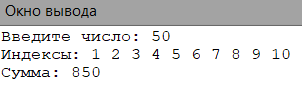
**end**

**end**;

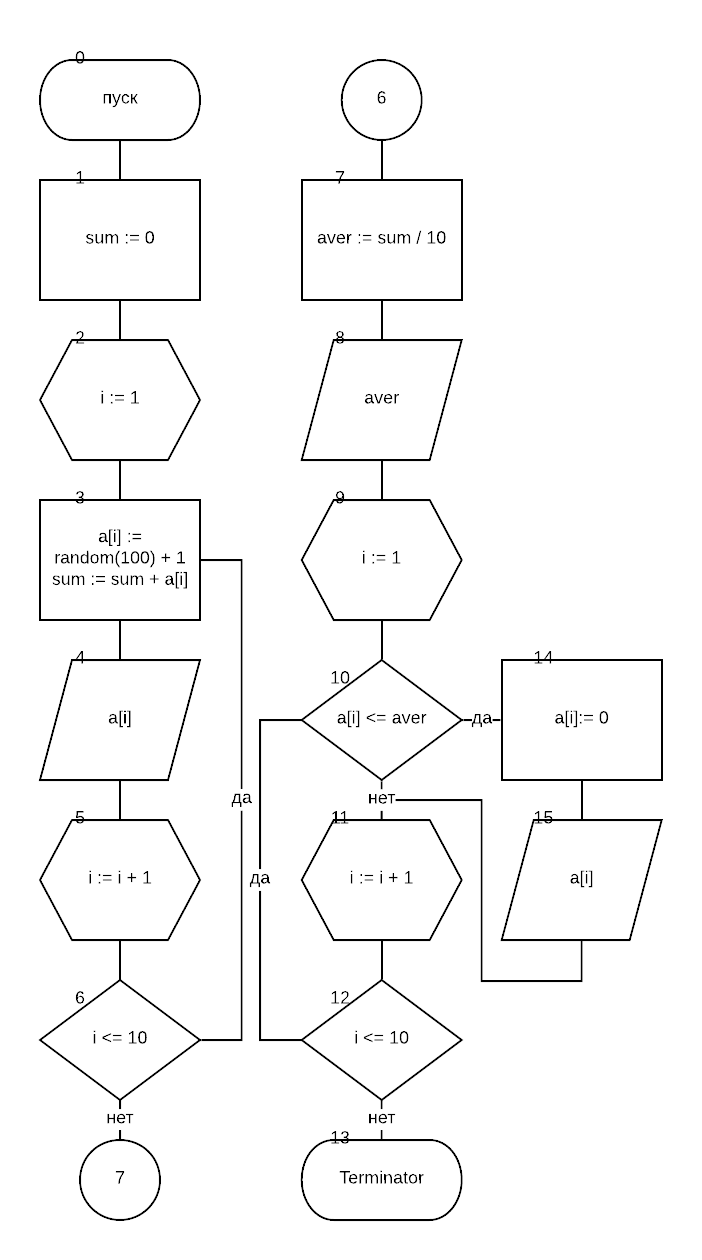
writeln();

writeln('Сумма: ', sum)

**end**.

1. 
2. Использовал цикл for для заполнения массива.
3. Нашёл сумму всех элементов массива, значения которых больше числа, введенного с клавиатуры, и вывел их индексы.

## 3 задание

1. Разветвляющиеся вычислительные процессы. Оператор выбора.
2. Научится реализовать алгоритм для вычисление разветвляющихся вычислительных процессов средствами компилятора и языка программирования Pascal.
3. ПК, PascalABC.NET 3.4.2.
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. aver := sum / 10.
6. 

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| i | Счётчик цикла | integer |
| sum | Сумма | integer |
| aver | Среднее арифметическое | real |

**program** pr10;

**var**

a: **array** [1..10] **of** integer;

i, sum: integer;

aver: real;

**begin**

randomize;

sum := 0;

write('Первый массив: ');

**for** i := 1 **to** 10 **do**

**begin**

a[i] := random(100) + 1;

sum := sum + a[i];

write(a[i], ' ')

**end**;

aver := sum / 10;

writeln();

writeln('Среднее арифметическое первого массива: ', aver);

write('Второй массив: ');

**for** i := 1 **to** 10 **do**

**begin**

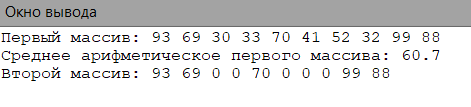
**if** a[i] <= aver **then**

a[i] := 0;

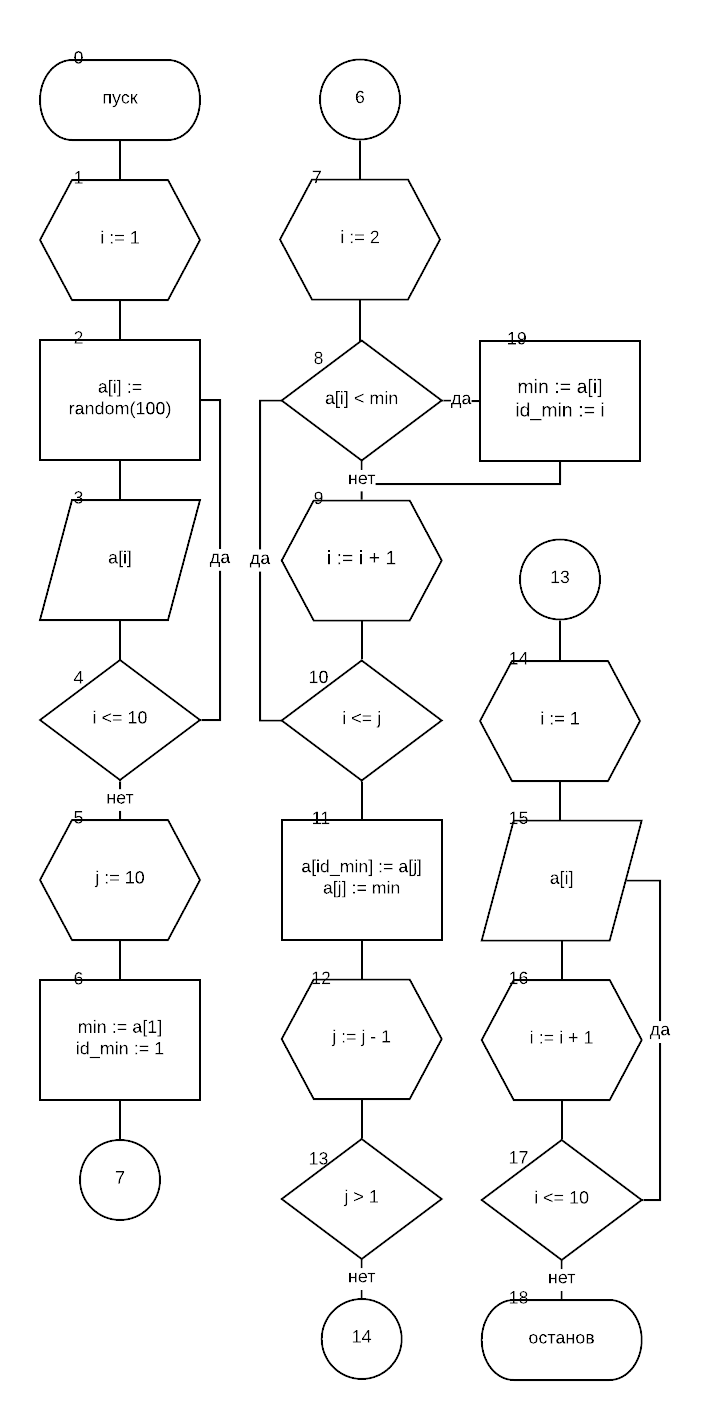
write(a[i], ' ')

**end**

**end**.

1. 
2. Сложил все элементы перового массива в одном цикле, потом нашёл их среднее арифметическое.
3. Дан одномерный массив. Нашёл его среднее арифметическое. Составил второй массив, элементами которого будут элементы первого массива, которые больше среднего арифметического. Остальные элементы заменил нулями.

## 4 задание

1. Разветвляющиеся вычислительные процессы. Оператор выбора.
2. Научится реализовать алгоритм для вычисление разветвляющихся вычислительных процессов средствами компилятора и языка программирования Pascal.
3. ПК, PascalABC.NET 3.4.2.
4. Дан одномерный массив. Упорядочить его по убыванию (см. материалы лекции).
5. min := a[i]; id\_min := i.
6. 

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| i | Счётчик цикла | integer |
| j | Счётчик цикла | integer |
| min | Минимальный элемент | integer |
| id\_min | Номер мин. элемента | integer |
| a | Массив | integer |

**program** pr10;

**var**

a: **array**[1..10] **of** integer;

i, j, id\_min, min: integer;

**begin**

randomize;

write('Исходный массив: ');

**for** i := 1 **to** 10 **do**

**begin**

a[i] := random(100);

write(a[i], ' ')

**end**;

writeln();

j := 10;

**while** j > 1 **do**

**begin**

min := a[1];

id\_min := 1;

**for** i := 2 **to** j **do**

**if** a[i] < min **then**

**begin**

min := a[i];

id\_min := i

**end**;

a[id\_min] := a[j];

a[j] := min;

j := j - 1

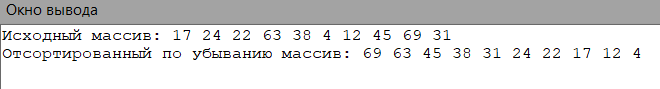
**end**;

write('Отсортированный по убыванию массив: ');

**for** i := 1 **to** 10 **do**

write(a[i], ' ')

**end**.

1. 
2. Программа сортирует массив методом выбора.
3. Упорядочил одномерный массив по убыванию.