



ОДНОМЕРНЫЕ МАССИВЫ ЦЕЛЫХ ЧИСЕЛ

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

9 класс

Ключевые слова

- массив
- описание массива
- заполнение массива
- вывод массива
- обработка массива
- последовательный поиск
- сортировка



Массив

Массив - это поименованная совокупность однотипных элементов, упорядоченных по индексам, определяющим положение элемента в массиве.

Одномерный массив

Значение элемента массива					Индекс элемента массива			
1	2	3	4	5	6	7	8	9
7	-5	9	1	0	-2	4	3	6

Решение разнообразных задач, связанных с обработкой массивов, базируется на решении таких типовых задач, как:

- суммирование элементов массива;
- поиск элемента с заданными свойствами;
- сортировка массива.

Описание массива

Общий вид описания массива:

```
var <имя_массива>: array [<мин_знач_индекса> ..  
<макс_знач_индекса>] of <тип_элементов>;
```

Имя массива

Тип элементов
массива

```
var a: array [1..10] of integer;
```

Минимальное значение
индекса

Максимальное значение
индекса

Значение 1-го
элемента
массива



```
const b: array [1..5] of integer = (4, 2, 3, 5, 7);
```

Массив **b** с постоянными значениями описан в разделе описания констант.

Способы заполнения массива

1 способ. Ввод каждого значения с клавиатуры:

```
for i:=1 to 10 do read (a[i]);
```



2 способ. С помощью оператора присваивания (по формуле):

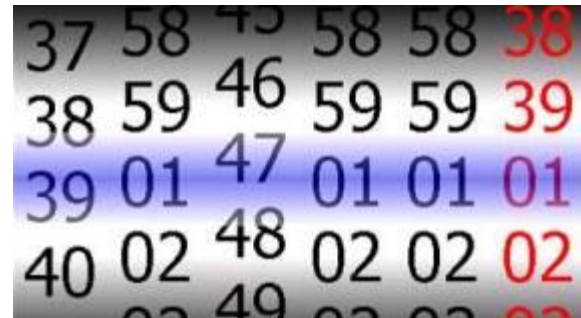
```
for i:=1 to 10 do a[i]:=i;
```

$a[i] = 2 * a[i] + 6$;

3 способ. С помощью оператора присваивания (случайными числами):

```
randomize;
```

```
for i:=1 to 10 do a[i]:=random(100);
```



Вывод массива

1 способ. Элементы массива можно вывести в строку, разделив их пробелом:

```
for i:=1 to 10 do write (a[i], ' ');
```

45 21 56 43 83 56 69 34 28 15

2 способ. Вывод с комментариями:

```
for i:=1 to 10 do writeln ('a[', i, ']=', a[i]);
```

a[1]=4

a[2]=1

a[3]=6

a[4]=3

a[5]=8

a[6]=5

a[7]=9

a[8]=4

a[9]=8

a[10]=7

Заполнение массива A(10) случайными числами и вывод элементов массива

Объявление
массива



Заполнение
массива



Вывод
массива

```
program n_2 ;  
  var i: integer;  
      a: array[1..10] of integer;
```

```
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);
```

```
    for i:=1 to 10 do write (a[i],` `);  
end.
```

Вычисление суммы элементов массива

Суммирование элементов массива осуществляется за счёт поочерёдного добавления слагаемых:

→ Определяется ячейка памяти (переменная s), в которой будет последовательно накапливаться результат суммирования

→ Переменной s присваивается начальное значение 0 - число, не влияющее на результат сложения

→ Для каждого элемента массива из переменной s считывается её текущее значение и складывается со значением элемента массива; полученный результат присваивается переменной s .

Вычисление суммы элементов массива

$s := 0$	$s = 0$
$s := s + a[1]$	$s = 0 + a[1]$
$s := s + a[2]$	$s = 0 + a[1] + a[2]$
$s := s + a[3]$	$s = 0 + a[1] + a[2] + a[3]$
...	...
$s := s + a[n]$	$s = 0 + a[1] + a[2] + a[3] + \dots + a[n]$

Основной фрагмент программы:

s:=0;

for i:=1 to n do s:=s+a[i];

Вычисление суммы элементов массива

```
program n_3;  
  var s, i: integer;  
    a: array[1..10] of integer;  
begin  
  s:=0;  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ` `);  
  for i:=1 to 10 do s:=s+a[i];  
  writeln ('s=', s)  
end.
```



Вычисление суммы элементов массива

```
program n_3_1;  
  var s, i: integer;  
      a: array[1..10] of integer;  
begin  
  s:=0;  
  randomize;  
  for i:=1 to 10 do  
    begin  
      a[i]:=random(100);  
      write (a[i], ' ');  
      s:=s+a[i]  
    end;  
  writeln ('s=', s)  
end.
```



Второй способ оформления:
Используем операторные скобки и всё (ввод, вывод, обработку) делаем в одном цикле.

Типовые задачи поиска

```
graph TD; A[Типовые задачи поиска] --> B[Нахождение наибольшего (наименьшего) элемента массива]; A --> C[Нахождение элемента массива, значение которого равно заданному значению];
```

Нахождение наибольшего (наименьшего)
элемента массива

Нахождение элемента массива, значение
которого равно заданному значению

Нахождение наибольшего элемента в стопке карточек с записанными числами:

- 1) Взять верхнюю карточку, записать на доске (запомнить) число как наибольшее.
- 2) Взять следующую карточку, сравнить числа. Если на карточке число больше, то стереть старую запись и записать это число.
- 3) Повторить действия, описанные в пункте 2 для всех оставшихся карточек.



При организации поиска
наибольшего элемента массива
правильнее искать его индекс.



Программа поиска наибольшего элемента в массива



```
program n_4;  
  var i, imax: integer;  
      a: array[1..10] of integer;  
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ' ');  
  imax:=1;  
  for i:=2 to 10 do  
    if a[i]>a[imax] then imax:=i;  
  write ('Наибольший элемент  a[, imax, ']=', a[imax])  
end.
```

Нахождение элемента массива с заданными свойствами

Результатом поиска элемента, значение которого равно заданному, может быть:

- n - индекс элемента массива такой, что $a[n] = x$, где x - заданное число;
- сообщение о том, что искомого элемента в массиве не обнаружено.

10	50	1	3	50	14	21	50	10	21
----	----	---	---	----	----	----	----	----	----

Здесь:

- трём равен 4-й элемент;
- десяти равны 1-й и 9-й элементы;
- нет элемента, равного 12.

Поиск элемента, равного 50

```
program n_5;  
  var n, i: integer;  
      a: array[1..10] of integer;  
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ` `);  
  n:=0;  
  for i:=1 to 10 do  
    if a[i]=50 then n:=i;  
    if n=0 then write('Нет') else write (i)  
  end.
```



Будет найден последний из элементов, удовлетворяющих условию.

10	50	1	3	50	14	21	50	10	21
----	----	---	---	----	----	----	----	----	----

Поиск элемента, равного 50

```
program n_5_1;  
  var n, i: integer;  
      a: array[1..10] of integer;  
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ' ');  
  i:=0;  
  repeat  
    i:=i+1;  
  until (a[i]=50) or (i=10);  
  if a[i]=50 then write(i) else write('Нет')  
end.
```

Будет найден первый из элементов, удовлетворяющих условию.

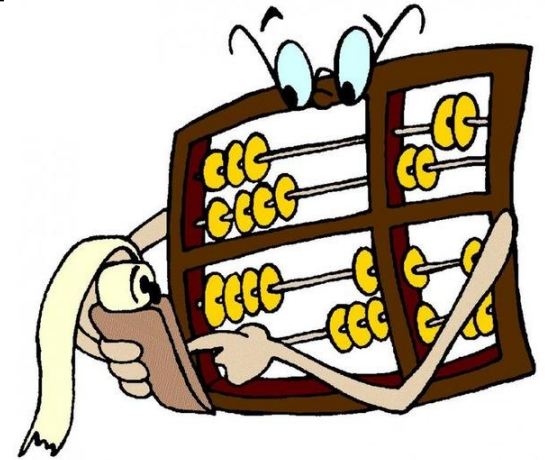


10	50	1	3	50	14	21	50	10	21
----	----	---	---	----	----	----	----	----	----

Подсчет количества элементов

Для подсчета вводится переменная, значение которой увеличивается на единицу каждый раз, когда найден нужный элемент.

```
program kolich;  
  var k, i: integer;  
    a: array[1..10] of integer;  
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ' ');  
  k:=0;  
  for i:=1 to 10 do if a[i]>50 then k:=k+1;  
  write('k=', k)  
end.
```



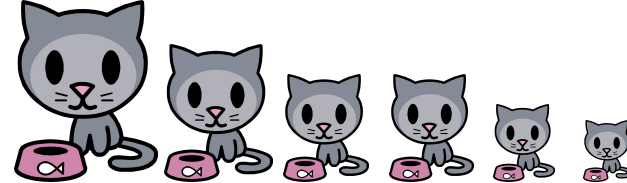
10	60	21	53	58	14	28	50	10	51
----	----	----	----	----	----	----	----	----	----

Сумма значений элементов, удовлетворяющих условию

```
program sum ;  
  var s, i: integer;  
      a: array[1..10] of integer;  
begin  
  randomize;  
  for i:=1 to 10 do a[i]:=random(100);  
  for i:=1 to 10 do write (a[i], ` `);  
  s:=0;  
  for i:=1 to 10 do  
    if (a[i]>10) and (a[i]<30) then s:=s+a[i];  
  write('s=', s)  
end.
```

10	50	1	3	50	14	21	50	10	21
----	----	---	---	----	----	----	----	----	----

Сортировка массива



Сортировка элементов массива по невозрастанию выбором осуществляется следующим образом:

1. В массиве выбирается максимальный элемент

2. Максимальный и первый элемент меняются местами (первый элемент считается отсортированным)

3. В неотсортированной части массива снова выбирается максимальный элемент; он меняется местами с первым неотсортированным элементом массива

Действия пункта 3 повторяются с неотсортированными элементами массива, пока не останется один неотсортированный элемент (минимальный)

Сортировка массива



Индекс		1	2	3	4	5	6	7	8
Значение		0	1	9	2	4	3	6	5
Шаги	1	0	1	9	2	4	3	6	5
	2	9	1	0	2	4	3	6	5
	3	9	6	0	2	4	3	1	5
	4	9	6	5	2	4	3	1	0
	5	9	6	5	4	2	3	1	0
	6	9	6	5	4	3	2	1	0
	7	9	6	5	4	3	2	1	0
	Итог:	9	6	5	4	3	2	1	0

Сортировка массива

0	1	9	2	4	3	6	5
---	---	---	---	---	---	---	---

```
program rost;  
  var n, i, j, x, imax: integer;  
      a: array [1..10] of integer;  
begin  
  for i:=1 to 10 do read (a[i]);  
  for i:=1 to 10 do write (a[i], ' ');  
  for i:=1 to 9 do  
    begin  
      imax:=i;  
      for j:=i+1 to 10 do if a[j]>a[imax] then imax:=j;  
      x:=a[i];  
      a[i]:=a[imax];  
      a[imax]:=x  
    end;  
  for i:=1 to 10 do write (a[i], ' ');  
end.
```

9	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

Самое главное

Заполнять массив можно либо вводя значение каждого элемента, с клавиатуры, либо присваивая элементам некоторые значения. При заполнении массива языках программирования используется цикл с параметром для реализации таких задач, связанных как с последовательностями и таблицами, так и с обработкой массивов, базируется на таких типовых задачах, как суммирование элементов массива, поиск элемента с заданными свойствами, сортировка массива.

```
var <имя массива>; array [<мин. знач. индекса> ..  
<макс. знач. индекса>] of тип_элементов;
```



Вопросы и задания

Запишите на языке Паскаль программу решения

задачи. Запишите программу, которая вычисляет среднюю за

неделю температуру воздуха. Исходные данные

задачи. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

температуре, количеству осадков, количеству дней в

неделе. Введите в программу значения, соответствующие

Опорный конспект

Массив - это поименованная совокупность однотипных элементов, упорядоченных по индексам, определяющим положение элементов в массиве.

var <имя_массива>: **array** [<мин_знач_индекса> .. <макс_знач_индекса>] **of** тип_элементов;

Заполнение массива

Ввод с клавиатуры

Присваивание значений

Задачи по обработке массива

Суммирование элементов

Сортировка элементов
массива

Поиск элементов массива с заданными
свойствами