

# Лекция

## **Работа с массивами**

# Теоретические сведения

- *Массив* – это упорядоченный набор данных одного типа, снабженных индексами.
- Индексы предназначены для нахождения определенного элемента массива, т.е. положение каждого элемента в массиве определяется его индексом.
- В этом и заключается упорядоченность.
- Тип компонент называется базовым типом массива.

В VBA массив рассматривается как переменная структурированного типа. Массиву присваивается имя, посредством которого можно ссылаться на него, как на единое целое, так и на любую из его компонент. Переменная с индексом — идентификатор компоненты массива. Формат записи:

**<имя массива> (<индекс>)**, где индекс может быть выражением порядкового типа.

Видим, что индексы принято указывать в круглых скобках после имени массива. По умолчанию нумерация элементов массива начинается с 0. Говорят, что 0 – *базовый индекс*.

Например, если A – массив из пяти чисел: 2, 3, 7, 9, 6, то  $A(0) = 2$  – первый элемент массива,  $A(4) = 6$  – последний.

Описание массива определяет имя, размер массива, базовый тип и производится в разделе переменных. В VBA имеются следующие способы описания массивов:

1. **<имя массива> (<номер последнего элемента>) [As <тип>]**
2. **<имя массива> (<начальный индекс> To <конечный индекс>) [As <тип>]**

Отметим, что второй способ позволяет изменить базовый индекс.

Другим способом изменения базового индекса является использование оператора **Option Base**, который имеет следующий синтаксис:

**Option Base <базовый индекс>**

**<базовый индекс>** – единица или ноль.

Этот оператор применяется перед процедурой, аналогично оператору **Option Explicit**.

Например, для изменения базового индекса с 0 на 1 используется оператор **Option Base 1**.

Массив А, который был рассмотрен, требует одного индекса для указания любого элемента.

Такой массив называется *одномерным (или линейным)* В одномерных массивах хранятся значения линейных таблиц. Примеры описания одномерных массивов:

1. Dim A (12) As Byte
2. Dim A (1 To 12) As Byte
3. Dim Bin (5) As Integer
4. Dim Str\_mass (4) As String
5. Dim K (7)
6. Dim L (0 To 3)

Заполнение массива в программе производится поэлементно.

Чаще всего для этого используется цикл с параметром, где в качестве параметра применяется индексная переменная.

Возможно заполнение массива путем простого присвоения значения элементам:

```
Dim B (1 To 3) As Integer
```

```
B (1) = 2
```

```
B (2) = 18
```

```
B (3) = 6
```

Удобным способом определения одномерных массивов является функция `Array`, преобразующая список элементов, разделенных запятыми, в вектор из этих значений, и присваивающая их переменной типа `Variant`.

```
Dim A As Variant
```

```
A = Array (10, 20, 35, 70)
```



Иногда в процессе выполнения программы требуется изменять размер массива. В этом случае первоначально массив объявляют как динамический. Для этого при объявлении массива не указывают его размерность. Например,

`Dim R () As Single`

В программе следует вычислить необходимый размер массива и связать его с некоторой переменной, например, `n`; затем изменить размер динамического массива с помощью оператора `ReDim`:

**`ReDim [Preserve] Имя (<номер последнего элемента>) [As <тип>]`**

**`ReDim [Preserve] Имя (<начальный индекс> To <конечный индекс>) [As <тип>]`**

`Preserve` – ключевое слово, используемое для сохранения данных в существующем массиве при изменении значения последней размерности.

*Двумерный массив* — структура данных, хранящая прямоугольную матрицу.

В матрице каждый элемент определяется номером строки и номером столбца, на пересечении которых он расположен.

В VBA двумерный массив может быть описан следующим образом:

1. **<имя массива> (<n1>, <n2>) [As <тип>]**

**<n1>, <n2>** – номер последнего элемента строки И номер последнего элемента строки соответственно;

2. **<имя массива> (<k1> To <k2>, <m1> To <m2>) [As <тип>]**

**<k1>, <k2>** – начальный и конечный индексы элементов строки;

**<m1>, <m2>** – начальный и конечный индексы элементов столбца.

Например,

**Dim mass (5, 5) As Integer**

**Dim mass (1 To 5, 1 To 5) As Integer**

описывают один и тот же массив, при условии, что был использован оператор Option Base 1.

Элементы двумерного массива идентифицируются переменными с двумя индексами.

Например:  $M(3, 5)$ .

Обычно первый индекс связывают с номером строки, второй — с номером столбца матрицы.

# Демонстрационные примеры:

1. Приведем фрагмент программы, осуществляющей ввод десяти элементов целочисленного массива A, формирование строки элементов массива и вывод его на экран.

```
Dim A(1 To 10) As Integer, i As Integer, Str As String
Str = " "
For i = 1 To 10
A(i) = Val(InputBox("Введите " & i & "-ый элемент
массива", "Заполнение массива"))
Str = Str & A(i) & " "
Next
MsgBox Str
```

2. Напишем программу, содержащую базовые элементы программ для обработки массивов:  
заполнение случайными числами динамического массива,  
поиск наибольшего и наименьшего элементов,  
перестановка элементов,  
дописывание в имеющийся массив новых элементов.

Option Explicit

Option Base 1

Sub Mass()

Dim M1() As Integer, M2() As Integer, n As Integer, i As Integer, max As Integer, min As Integer, \_

Str1 As String, Str2 As String, Str3 As String, sum As Integer, pro As Single, buf As Integer

n = InputBox("Введите количество элементов массива",  
"Определение размера массива")

ReDim M1(n), M2(n)

'Заполнение массива случайными числами в диапазоне от 1 до 10

'и формирование строки значений элементов массива

Randomize

For i = 1 To n

M1(i) = Int(10 \* Rnd + 1)

M2(i) = Int(10 \* Rnd + 1)

Str1 = Str1 & M1(i) & " "

Str3 = Str3 & M2(i) & " "

Next

# 'Поиск максимального элемента массива M1

max = M1(1)

For i = 2 To n

If M1(i) > max Then max = M1(i)

Next



# 'Поиск минимального элемента массива M1

min = M1(1)

For i = 2 To n

If M1(i) < min Then min = M1(i)

Next

'Поиск суммы элементов массива  
M1, стоящих на четных местах

sum = 0

For i = 2 To n Step 2

sum = sum + M1(i)

Next

# 'Поиск произведения ненулевых элементов массива M1

pro = 1

For i = 1 To n

If M1(i) <> 0 Then pro = pro \* M1(i)

Next

# 'Поменяем местами 1-ый и 2-ой элементы массива M1

```
If n >= 2 Then
```

```
  buf = M1(1)
```

```
  M1(1) = M1(2)
```

```
  M1(2) = buf
```

```
End If
```

```
For i = 1 To n
```

```
  Str2 = Str2 & M1(i) & " "
```

```
Next
```

```
MsgBox "Массив: " & Str1 & Chr(13) &  
    "Максимальный элемент: " & max & Chr(13) &  
    _  
    "Минимальный элемент: " & min & Chr(13) &  
    "Сумма элементов массива, стоящих на  
    четных местах: " _  
    & sum & Chr(13) & "Произведение ненулевых  
    элементов массива: " & pro & Chr(13) _  
    & "Массив после обмена 1-го и 2-го элементов: "  
    & Str2
```

# 'Допишем в массив M2 максимальный и минимальный элементы массива M1

```
ReDim Preserve M2(n + 2)
```

```
M2(n + 1) = max
```

```
M2(n + 2) = min
```

```
Str2 = ""
```

```
For i = 1 To n + 2
```

```
Str2 = Str2 & M2(i) & " "
```

```
Next
```

```
MsgBox "Первый массив: " & Str1 &  
Chr(13) & "Второй массив: " & Str3 &  
Chr(13) & _
```

```
"Второй массив с приспанным  
максимумом и минимумом из первого: "  
& Chr(13) & Str2
```

```
End Sub
```

3. Дан целочисленный линейный массив.  
Отсортировать его в порядке убывания значений.

Воспользуемся алгоритмом, известным под названием “метод пузырька”.

Идея состоит в последовательном перемещении путем попарных перестановок наибольшего элемента вначале на место  $n$ -го элемента, затем  $n-1$ -го элемента и т.д.



Option Explicit

Option Base 1

Sub Сортировка()

Dim A() As Integer, n As Integer, i As Integer, j As Integer, P As Integer, Str1 As String, Str2 As String

n = InputBox("Введите число элементов массива:", "Определение размера")

ReDim A(n) As Integer

Str1 = ""

Str2 = ""

# 'Заполнение массива с клавиатуры и формирование строки

```
For i = 1 To n
```

```
A(i) = Val(InputBox("Введите " & i & "-ый  
элемент массива", "Заполнение  
массива"))
```

```
Str1 = Str1 & A(i) & " "
```

```
Next
```

# 'Вложенные циклы, организующие сортировку массива

```
For i = 1 To n - 1  
  For j = 1 To n - 1  
    If A(j) <= A(j + 1) Then  
      P = A(j)  
      A(j) = A(j + 1)  
      A(j + 1) = P  
    End If  
  Next  
Next
```

# 'Формирование строки, содержащей элементы отсортированного массива

```
For i = 1 To n
```

```
Str2 = Str2 & A(i) & " "
```

```
Next
```

```
MsgBox "Исходный массив: " & Str1 &  
Chr(13) & "Отсортированный массив: " &  
Str2, , "Результат"
```

```
End Sub
```

4. Пример заполнения двумерного динамического массива случайными целыми числами и вывод на экран.

Option Explicit

Option Base 1

Sub Массив\_двумерный ()

Dim mass() As Integer, i As Integer, j As Integer, n  
As Integer, m As Integer, str As String

n = InputBox("Введите количество строк  
массива", "Размер массива")

m = InputBox("Введите количество столбцов  
массива", "Размер массива")

'Переопределение размерности массива

ReDim mass(1 To n, 1 To m) As Integer

Randomize

# Заполнение массива случайными значениями

'и накопление элементов массива для последующего вывода значений

```
For i = 1 To n
For j = 1 To m
mass(i, j) = Int(100 * Rnd + 1)
str = str & mass(i, j) & "  "
Next j
str = str & Chr(13)
Next i
MsgBox "Массив: " & Chr(13) & str, , "Результат"
End Sub
```

Сформировать матрицу Пифагора (таблицу умножения в матричной форме) и вывести ее на экран.

Значения элементов матрицы Пифагора вычисляются следующим образом:

$$P(i, j) = i * j.$$

Вычисления и вывод матрицы производятся в двух вложенных циклах. Вывод на экран организуем в виде прямоугольной таблицы.



```
Option Explicit
Option Base 1
Sub Пифагор()
Dim P(1 To 9, 1 To 9) As Integer, i As Integer, j As Integer,
    Str As String
Str = ""
For i = 1 To 9
For j = 1 To 9
P(i, j) = i * j
Str = Str & P(i, j) & "  "
Next
Str = Str & Chr(13)
Next
MsgBox "Матрица Пифагора: " & Chr(13) & Str, ,
    "Результат"
End Sub
```

**Пример.** Найти сумму элементов матрицы  $A(3,2)$

```
Public Sub СуммаМатрицы()  
Dim I, J As Integer  
Dim A(3, 2), S As Single  
S = 0  
' Цикл перебора строк  
For I = 1 To 3  
    ' Цикл перебора столбцов  
    For J = 1 To 2  
        ' Добавление слагаемого  
        S = S + A(I, J)  
        ' Конец цикла по столбцам  
    Next  
    ' Конец цикла по строкам  
Next  
End Sub
```

**Пример.** Найти произведение положительных и отрицательных элементов матрицы  $M(6,12)$

```
Public Sub ПлюсМинус()  
Dim I, J As Integer  
Dim M(6, 12), Рплюс, Рминус As Single  
Рплюс = 1  
Рминус = 1  
For I = 1 To 6  
    For J = 1 To 12  
        If M(I, J) > 0 Then  
            Рплюс = Рплюс * M(I, J)  
        ElseIf M(I, J) < 0 Then  
            Рминус = Рминус * M(I, J)  
        End If  
    Next  
Next  
MsgBox ("Произведение положительных" & Рплюс)  
MsgBox ("Произведение отрицательных" & Рминус)  
End Sub
```

## Контрольный пример

Сколько чисел будет просуммировано в результате работы приведенного фрагмента программы?

S = 0

For I = 1 To 5

For J = 1 To 4

S = S + D(I,J)

Next

Next

D

	1	2	3	4	
1	1	2	3	4	10
2	1	4	7	8	20
3	1	6	11	12	30
4	1	8	15	16	40
5	1	10	19	20	50
	5	30	55	60	

Варианты ответов:

A. 5

B. 4

C. 20

D. 9

## Контрольный пример

Чему равно  $S$  в результате работы приведенного фрагмента программы?

$S = 0$

For I = 1 To 5

For J = 1 To 4

$S = S + D(I,J)$

Next

Next

D

	1	2	3	4	
1	1	2	3	4	10
2	1	4	7	8	20
3	1	6	11	12	30
4	1	8	15	16	40
5	1	10	19	20	50

5      30      55      60

Варианты ответов:

A. 10

B. 150

C. 60

D. 50

## Контрольный пример

Чему равно  $S$  в результате работы приведенного фрагмента программы?

```
For I = 1 To 5
  S = 0
  For J = 1 To 4
    S = S + D(I,J)
  Next
Next
```

D

	1	2	3	4	
1	1	2	3	4	10
2	1	4	7	8	20
3	1	6	11	12	30
4	1	8	15	16	40
5	1	10	19	20	50
	5	30	55	60	

Варианты ответов:

- A. 10
- B. 150
- C. 60
- D. 50

## Контрольный пример

Какой результат вычисляет приведенный фрагмент программы?

Варианты ответов:

```
P = 1
```

```
For I = 1 To 7
```

```
  For J = 1 To 5
```

```
    P = P*A(I,J)
```

```
  Next
```

```
Next
```

A. Суммы индексов элементов пятой строки.

B. Произведения элементов 7 –ой строки.

C. Сумма всех элементов.

D. Произведение всех элементов

## Контрольный пример

Какой результат вычисляет приведенный фрагмент программы?

```
S = 0
```

```
For I = 1 To 3
```

```
  For J = 1 To 5
```

```
    S = S + C(3,J)
```

```
  Next
```

```
Next
```

Варианты ответов:

A. Сумму элементов третьего столбца.

B. Сумму элементов третьей строки.

C. Сумму всех элементов.

D. Утроенную сумму элементов третьей строки.

.



## Контрольный пример

Какой результат вычисляет приведенный фрагмент программы?

```
S3 = 0
```

```
For I = 1 To 3
```

```
  For J = 1 To 5
```

```
    S = S3 + C(3,J)
```

```
  Next
```

```
Next
```

Варианты ответов:

A. Сумму элементов третьего столбца.

B. Сумму элементов третьей строки.

C. Сумму всех элементов.

D. A,B,C - не верны.

.

## Контрольный пример

Какой результат вычисляет приведенный фрагмент программы?

```
P = 1
```

```
For I = 1 To 7
```

```
For J = 1 To 5
```

```
P = P*A(I,J)
```

```
Next
```

```
Next
```

Варианты ответов:

- A. Суммы всех строк.
- B. Произведения всех столбцов.
- C. Сумма всех элементов.
- D. Произведение всех элементов

**Пример.** Массив D расположен в ячейках A2:F5 ЭТ с именем **Массив**. Найти произведения его элементов по столбцам и записать их в массив с именем P. Результат поместить на лист **Произведение**, в ячейки D1:D6.

### Решение

1. Исходный массив:

	A	B	C	D	E	F
1						
2	X	X	X	X	X	X
3	X	X	X	X	X	X
4	X	X	X	X	X	X
5	X	X	X	X	X	X



Массив D(4,6)

2. Результат:

	A	B	C	D
1				X
2				X
3				X
4				X
5				X
6				X



Массив P(6)

```

Public Sub УмножениеСтолбцов()
Dim D(4, 6), P(6), I, J As Integer
Dim Массив, Произведение As Object
Set Массив = Sheets("Массив")
Set Произведение = Sheets("Произведение")
' считывание массива из ЭТ
For I = 1 To 4
For J = 1 To 6
C(I, J) = Массив.Cells(I + 1, J)
Next
Next
' Произведения по столбцам
For J = 1 To 6
P(J) = 1
For I = 1 To 4
P(J) = P(J) * C(I, J)
Next
' Запись в ЭТ
Произведение.Cells(J, 4) = P(J)
Next
End Sub

```

## Контрольный пример

Сколько чисел будет получено в результате работы приведенного фрагмента программы?

```
For I = 1 To 7
```

```
P(I) = 1
```

```
For J = 1 To 5
```

```
P(I) = P(I)*N(I,J)
```

```
Next
```

```
Next
```

Варианты ответов:

A. 5

B. 7

C. 35

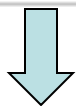
D. 12

**Пример.** Массив К расположен в ячейках C4:D8 листа **Первый**. Записать удвоенное значение элементов этого массива в ячейки B2:C6 листа **Второй**.

Исходный массив

	A	B	C	D
1				
2				
3				
4			X	X
5			X	X
6			X	X
7			X	X
8			X	X

→ K(5,2)



Четвертая строка,  
Третий столбец

Результат вычислений

	A	B	C
1			
2		X	X
3		X	X
4		X	X
5		X	X
6		X	X



Вторая строка, второй  
столбец

```
Public Sub УдвоениеМассива()  
Dim I, J As Integer  
Dim K(5, 2) As Single  
Dim Первый, Второй As Object  
Set Первый = Sheets("Первый")  
Set Второй = Sheets("Второй")  
    ' Запись в массив с листа Первый  
For I = 1 To 5  
For J = 1 To 2  
K(I, J) = Первый.Cells(I + 3, J + 2)  
Next  
Next  
    ' Запись результата на лист Второй  
For I = 1 To 5  
For J = 1 To 2  
Второй.Cells(I + 1, J + 1) = 2 * K(I, J)  
Next  
Next  
End Sub
```

## Контрольный пример

Укажите адрес ячейки, начиная с которой осуществляется считывание массива М из ЭТ?

```
For I = 1 To 5
```

```
  For J = 1 To 8
```

```
    M(I,J) = Старый.Cells(I+3,J+1)
```

```
    Новый.Cells(I,J+4) = 3*M(I,J)
```

```
  Next
```

```
Next
```

Варианты ответов:

А. Лист Старый, ячейка В4.

В. Лист Старый, ячейка D2.

С. Лист Новый, ячейка В4.

Д. Лист Новый, ячейка D2.



## Контрольный пример

Укажите адрес ячейки, начиная с которой осуществляется запись массива Р в ЭТ?

```
For I = 1 To 7
```

```
  For J = 1 To 6
```

```
    P(I,J) = Третий.Cells(I+4,J)
```

```
    Четвертый.Cells(I+6,J+3) = 5+P(I,J)
```

```
  Next
```

```
Next
```

Варианты ответов:

А. Лист Третий, ячейка А5.

В. Лист Третий, ячейка Е1.

С. Лист Четвертый, ячейка D7.

Д. Лист Четвертый, ячейка G4.