

МАССИВЫ

Одномерные массивы

Определение массива

Массив – совокупность однотипных данных.

Массивы

```
graph TD; A[Массивы] --> B[Числовые:]; A --> C[Символьные:];
```

Числовые:

100

0.345

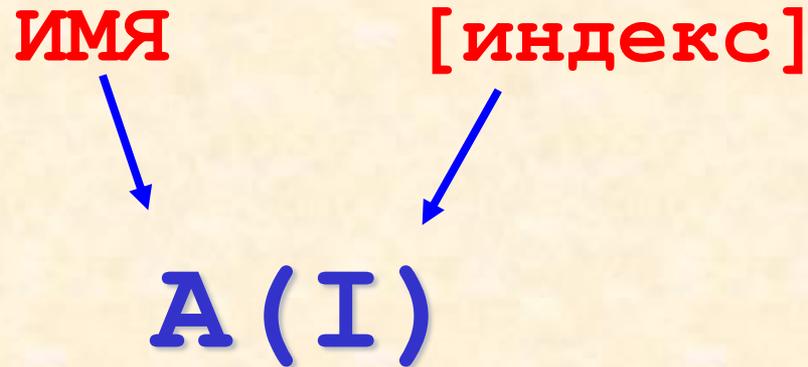
-34,7

Символьные:

Дом , Student

Характеристики массива

- Имя
- Размерность
- Размер



Имя массива

- *это набор букв латинского алфавита*

Например: A, L, D, F, RT, NQ, MAR

Так же, может быть сочетание букв и цифр,
например, A1, M23, BP4

Размерность массива

По размерности массивы различаются на *одномерные*, *двумерные* и *многомерные*

- Одномерные массивы представляют собой линейку или столбец символов

15	0	25	17	-6	-12
----	---	----	----	----	-----

- Двумерные массивы представляют собой таблицу

7	9	6	-2	7
-8	19	12	14	23
16	-15	19	-9	20

- Многомерные (трехмерные) массивы представляют собой набор таблиц

Размер одномерного массива

- определяет количество элементов в массиве.

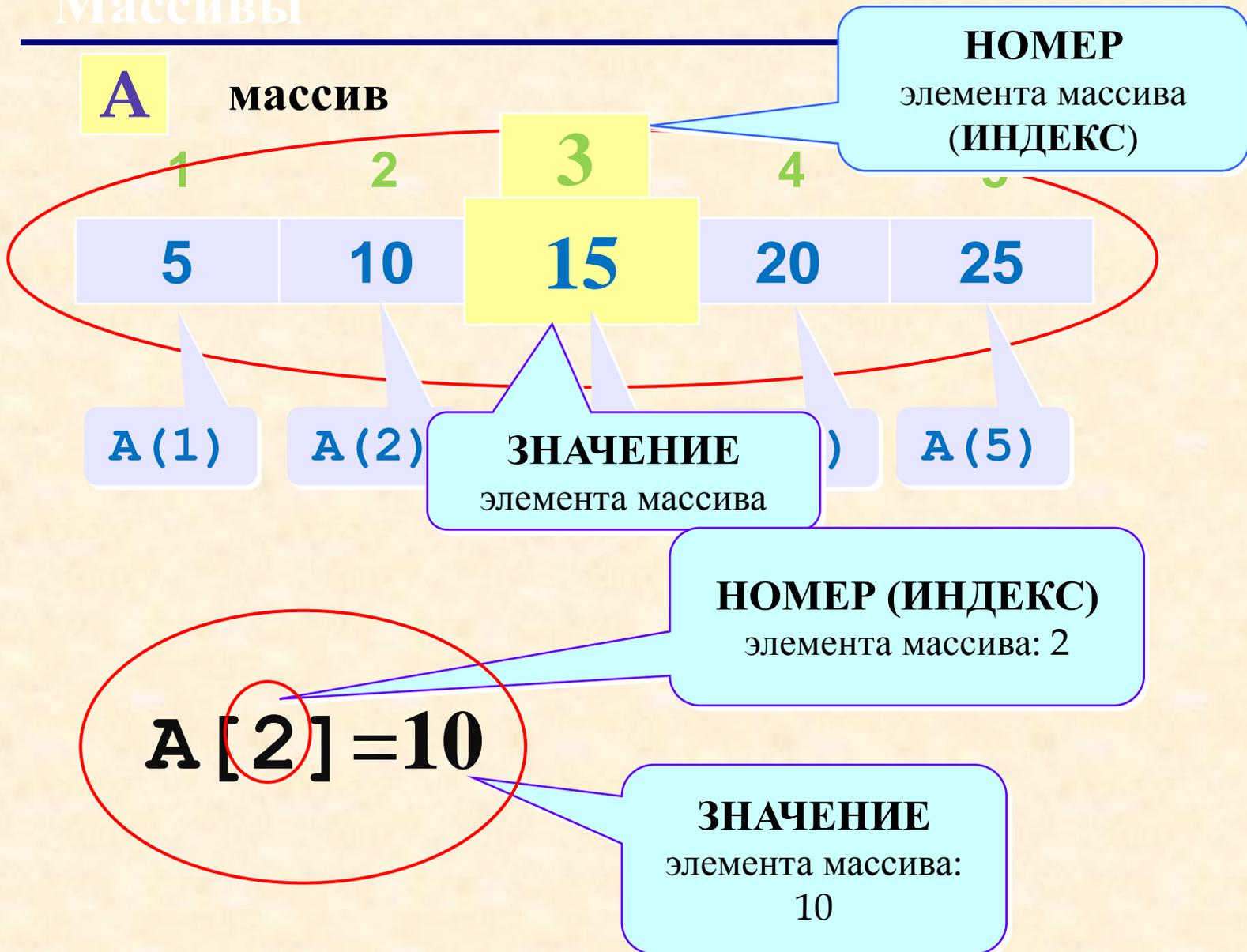
15	0	25	17	-6	-12
----	---	----	----	----	-----

A (1..6) – массив A содержит 6 элементов:

номер первого элемента – 15;

номер последнего элемента – -12.

Массивы



Объявление массивов

Для объявления массивов используется оператор

Dim (от англ. Dimension – размерность).

С помощью **Dim** можно объявлять как одномерные, так и многомерные массивы.

Объявление массива с помощью оператора **Dim** имеет следующий синтаксис:

Dim имя массива (*[измерения массива]*) [as type]

Например:

Dim A () As Integer

Dim S (10) As Integer

Dim Array(1 to 100) As String

Dim Luna (1 To 15) As String, Alfa (1 To 10) As
Long

ВВОД ЭЛЕМЕНТОВ МАССИВА

(Заполнение массивов)

Массив позволяет:

1)сохранять и манипулировать многими элементами данных посредством единственной переменной. (пример)

2)можно использовать циклы для легкой обработки различных элементов массивов. Объединяя массивы и структуры цикла (обычно **For...Next**), можно написать небольшое число операторов, которые обрабатывают большой объем данных. (Пример, ниже)

Ввод массива в VBA может осуществляться:

- с клавиатуры;
- считывать данные с листа Excel;
- с помощью датчика случайных чисел;
- с помощью формулы.

1) Функция **InputBox** (отображает диалоговое окно, содержащее текст, который запрашивает пользователя ввести некоторое значение и текстовое окно для ввода этого значения). Первый вариант рекомендуется использовать при вводе небольших по размеру массивов. Ввод каждого элемента массива осуществляется непосредственным набором цифр на клавиатуре. Ошибочно введенное число записывается в память ЭВМ и для исправления его нужно начать выполнять программу с самого начала, что вовсе не гарантирует от появления новой ошибки при вводе.

Например:

Sub primer()

Dim A(5) As Integer

For i=1 To 5

**A(i) = InputBox("Введите элемент массива A(" &
i & ")", "Заполнение массива")**

Next i

.....

[продолжение программы]

При своей работе приведенный фрагмент программы выводит следующие окна:

Заполнение массива

Введите элемент массива A(1)

OK

Cancel

24

Заполнение массива

Введите элемент массива A(2)

OK

Cancel

4

Заполнение массива

Введите элемент массива A(3)

OK

Cancel

54

Заполнение массива

Введите элемент массива A(4)

OK

Cancel

8

Заполнение массива

Введите элемент массива A(5)

OK

Cancel

-9

2) Функция **Cells**— данные считываются из рабочего листа Excel. В этом случае пользователь перед запуском программы вводит данные элементов массива в ячейки рабочего листа Excel. Этот способ более предпочтителен, поскольку он не требует ввода данных с клавиатуры каждый раз при новом появлении диалогового окна.

Например:

Sub primer()

Dim A(5) As Integer

For i=1 To 5

**A(i)=Cells(1, i) ‘ считывание значения из 1– ой
строки, i –го столбца**

Next i

.....

[продолжение программы]

Примечание: Номер индекса в примере совпадает с номером
столбца

Результат выполнения фрагмента программы

	A	B	C	D	E	F	G	H
1	4	6	-15	8	24			
2								
3								
4	A(1)	A(2)	A(3)	A(4)	A(5)			
5								
6								

The diagram shows a spreadsheet with columns A through H and rows 1 through 6. Row 1 contains the values 4, 6, -15, 8, and 24 in columns A through E respectively. Row 4 contains callouts A(1), A(2), A(3), A(4), and A(5) in columns A through E respectively. Green callout boxes with pointers connect the callouts to the corresponding cells in row 4. A black rectangle highlights the cell A2.

3) Использование генератора случайных чисел **Rnd()**, если значение массива не задано. При таком способе заполнения массива надо учесть, что перед использованием функции **Rnd()** надо инициализировать генератор случайных чисел, а также то, что он выдает числа в интервале от 0 до 1.

Sub primer()

Randomize Timer

Dim A(5) As Integer

For i=1 To 5

A(i)=Rnd()

Next i

.....

[продолжение программы]

4) с помощью формулы, если массив заполняется по какой-либо закономерности, причем значение элемента массива зависит от его местоположения:

```
Dim a(5) As Single
```

```
For i = 1 To 5
```

```
a(i) = 5 * i / (i + 4)
```

```
Next i
```

.....

[продолжение программы]

ВЫВОД МАССИВОВ

1) в рабочий лист Excel – для этого используется функция **Cells**:

.....

For i = 1 To n

Cells(10, i) = a(i)

Next I

.....

вывод в 10 строку рабочего листа

7							
8							
9							
10	4	6	-15	8	24		
11							
12							
13							

2) в диалоговое окно. Для этого весь массив переписывается в какую-либо одну текстовую переменную (например Text), которая и выводится в диалоговое окно. Функция Chr(13) – перевод каретки, которая при заполнении строки переводит курсор на начало следующей строки и **Chr(9)** – **вставляет символ табуляции** между элементами массива в строке.

Например:

.....

Text = ""

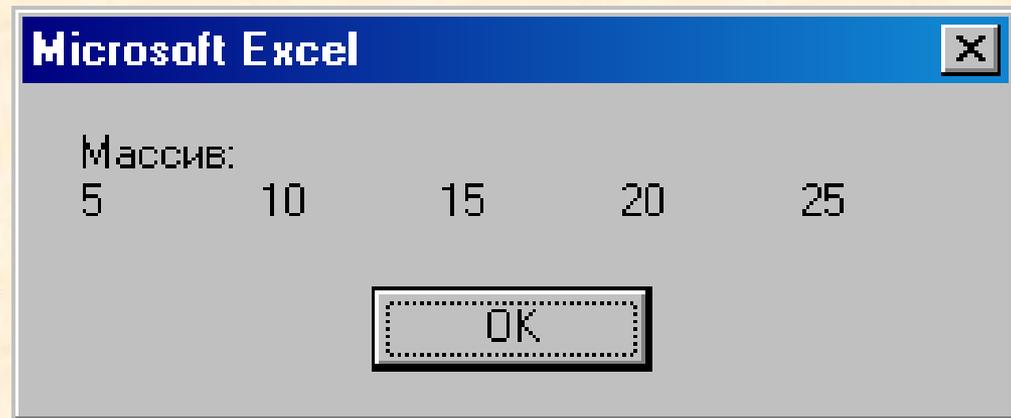
For i = 1 To 5

Text = Text & a(i) & Chr(9)

Next I

**MsgBox "Массив:" & Chr(13) &
Text**

.....



ДИНАМИЧЕСКИЕ МАССИВЫ

Динамические массивы используются в том случае, когда количество элементов массива заранее неизвестно и будет определяться в процессе выполнения программы. После того, как работа с динамическим массивом в программе будет выполнена, можно освободить память, которую он занимает.

Динамический массив объявляется так же, как и любой другой тип массива — оператором **Dim**. Единственное отличие состоит в том, что вы не указываете (в скобках) размер массива.

Например:

Dim A() As Byte ‘ объявление динамического массива с указанием типа массива

Этой строкой мы создаем массив, но откладываем решение о том, сколько он будет иметь элементов. Перед тем, как использовать этот массив, нам необходимо будет указать его размер, который во время работы программы мы можем изменять сколько угодно раз. Это делается с помощью оператора **ReDim**.

Например:

ReDim A(5) ‘ указание размерности динамического массива

При указании значений для размерностей динамического массива оператором **ReDim** его тип не может быть изменен. В противном случае будет выдано сообщение об ошибке.

При помощи оператора **ReDim** можно установить любые значения границ и количество размерностей, а также менять любой из установленных параметров как в сторону уменьшения, так и увеличения.

Например:

Dim A() As Byte

ReDim A(1 to 4)

‘ Диапазон от 1 до 4

В качестве как верхней, так и нижней границы могут быть использованы не только числа, но и переменная целого типа, например:

Dim n As Integer

Dim A() As Integer

N=10

ReDim A(n)

При каждом выполнении оператора **ReDim** все значения элементов массива, которые до этого в нем хранились, будут потеряны, так как данный оператор обнуляет все элементы в соответствии с их типом, например, числовые массивы принимают значение 0, а символьные “” (пусто).

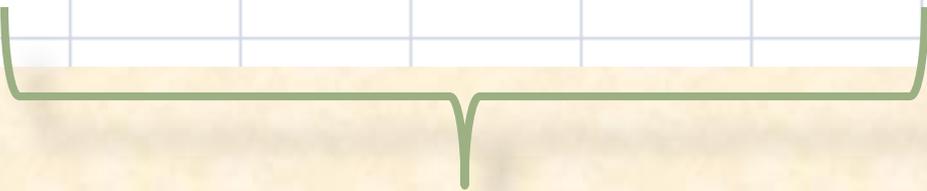
Для того, чтобы имеющиеся в массиве значения не пропали при его переопределении, следует использовать служебное слово **Preserve**

Например:

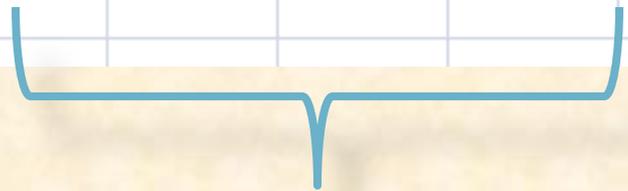
```
Sub task_1()  
Dim n As Integer  
Dim i As Integer  
Dim A() As Integer  
n = 6  
ReDim A(n)  
For i = 1 To n  
    A(i) = i ^ 2  
Next i  
n = 10  
ReDim Preserve A(n)  
For i = 7 To n  
    A(i) = i ^ 3  
Next i  
For i = 1 To n  
    Cells(1, i) = A(i)  
Next i  
End Sub
```

В данной программе требуется заполнить массив из 6 элементов квадратами первых 6 натуральных чисел. Затем следует переопределение размера массива до 10 и заполнение новых элементов (с 7 по 10) кубами следующих натуральных чисел.

	A	B	C	D	E	F	G	H	I	J
1	1	4	9	16	25	36	343	512	729	1000
2										
3										



$$i^2$$



$$i^3$$

Пример заполнения массива A(N) с помощью Rnd

```
Sub task()
```

```
Dim n As Integer
```

```
Dim i As Integer
```

```
Dim A() As Integer
```

```
n = InputBox ("Введите размер массива", «Заполнение массива»)
```

```
ReDim A(n) As Integer
```

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

```
    A(i) = Rnd() * 100 - 20
```

```
Next I
```

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

```
    Cells(1, i) = A(i)
```

```
Next i
```

```
End Sub
```


ДВУМЕРНЫЕ МАССИВЫ

Любая таблица (матрица) чисел или символов по определению является *двумерным массивом*. Двумерный массив $A(N, M)$ соответствует понятию прямоугольной матрицы (таблицы), состоящей из N строк и M столбцов.

В двумерном массиве каждый элемент фиксируется номером строки и столбца, на пересечении которых он расположен.

Положение элемента в двумерном массиве определяется двумя индексами. Они разделяются запятой.

В качестве индексов могут использоваться любые арифметические выражения, константы, переменные. Если арифметическое выражение содержит дробную часть, то при определении индекса она отбрасывается.

При появлении отрицательного значения индекса вызывает сообщение об ошибке.

При обращении к элементу двумерного массива необходимо указывать два индекса: номер строки и номер столбца.

При $N=3$ и $M=3$

$a[1,1]$ $a[1,2]$ $a[1,3]$

$a[2,1]$ $a[2,2]$ $a[2,3]$

$a[3,1]$ $a[3,2]$ $a[3,3]$

Номер
строки

Номер
столбца

В редакторе VBA двумерный массив задается с помощью оператора **Dim** (если N и M конкретные числа):

Dim A(10,5) As Integer

Если N и M переменные величины то массив объявляется как динамический:

ReDim A(N,M) As Integer

Здесь оператором **ReDim** объявляется двумерным массив именем A размерностью $N \times M$,

где

N - количество строк (первый индекс массива),
 M – количество столбцов (второй индекс). Значения переменных N и M , определяющие размер массива, должны быть заданы до оператора **ReDim**. Сам оператор **ReDim** должен находиться до первого обращения к элементам массива, т.е. как правило – в начале программы.

В качестве имени массива можно использовать любое допустимое на языке VBA имя переменной. В одном операторе **Dim** можно объявить несколько массивов. Можно также использовать несколько операторов **Dim**.

Dim A(10,5) As Integer, R(2,3) As Long

ВВОД, ВЫВОД ЭЛЕМЕНТОВ ДВУМЕРНОГО МАССИВА

Поскольку элементы двумерного массива задаются двумя индексами – номером строки и номером столбца – для перебора всех строк и столбцов необходимо использование двойного цикла, один из которых будет внешним, а другой – внутренним. Такие циклы называются ***вложенными***.

1)С помощью функции `InputBox`.

```
Sub primer()  
Dim A (3,4) As Integer  
    For i = 1 To 3    ' цикл по строкам  
        For j = 1 To 4    ' цикл по столбцам  
            A(i, j) = InputBox("Введите  
                элемент массива")  
        Next j  
    Next i
```

.....

[продолжение программы]

2) С помощью функции **Cells**:

```
Sub primer()
```

```
Dim A (3,4) As Integer
```

```
For i = 1 To 3 ' цикл по строкам
```

```
    For j = 1 To 4 ' цикл по столбцам
```

```
        A(i, j) = Cells (I, J)
```

```
    Next j
```

```
Next i
```

.....

[продолжение программы]

Этот способ наиболее удобен (данные не теряются при повторном выполнении программы).

3) Функция RND (если значение массива не задано):

```
Sub task ()
```

```
Dim A ()
```

```
N = 3: M = 4
```

```
ReDim A(N, M) As Single
```

```
For i = 1 To N
```

```
    For j = 1 To M
```

```
        A(i, j) = Rnd() * 100
```

```
    Next j
```

```
Next i
```

.....

[продолжение программы]

Вывод двумерного массива так же как и ввод производится с использованием двойного цикла. В зависимости от условий задачи **ввод и вывод можно организовывать по строкам или по столбцам**. При вводе и выводе по строкам параметром внешнего цикла является индекс строки, параметром внутреннего цикла – индекс столбца. При вводе по столбцам – внешний цикл по столбцам, внутренний – по строкам.

- 1) в рабочий лист Excel – для этого используется известная уже функция **Cells**:

□ ...

- **For i = 1 To n**
 - **For j = 1 To m**
 - **Cells(i, j) = a(i, j)**
 - **Next j**
- **Next i**

□ ...

2) в диалоговое окно. Для этого весь массив переписывается в какую-либо одну текстовую переменную (например Text), которая и выводится в диалоговое окно.

...

- ▣ **Text = ""**
- ▣ **For i = 1 To n**
 - ▣ **For j = 1 To m**
 - ▣ **Text = Text & a(i, j) & Chr(9)**
 - ▣ **Next j**
 - ▣ **Text = Text & Chr(13)**
- ▣ **Next i**
- ▣ **MsgBox "Массив:" & Chr(13) & Text**

▣ ...

- ▣ Примечание: Chr(13)- перевод каретки, которая при заполнении строки переводит курсор на начало следующей строки и **Chr(9)** – вставляет символ табуляции между элементами массива в строке.

Операции с двумерными массивами (матрицами)

Двумерные массивы, у которых значения индексов одинаковые, т.е. число строк равно числу столбцов, называют *квадратной матрицей*.

В зависимости от соотношения между первым и вторым индексами элемента определяется расположение этого элемента в матрице.

Побочная диагональ

$$j = n - i + 1$$

$$\mathbf{B} = \begin{array}{|cccc|} \hline \mathbf{b}_{1=1} & \mathbf{b}_{1<2} & \mathbf{b}_{13} & \mathbf{b}_{14} \\ \mathbf{b}_{21} & \mathbf{b}_{22} & \mathbf{b}_{23} & \mathbf{b}_{24} \\ \mathbf{b}_{3>1} & \mathbf{b}_{32} & \mathbf{b}_{33} & \mathbf{b}_{34} \\ \mathbf{b}_{41} & \mathbf{b}_{42} & \mathbf{b}_{43} & \mathbf{b}_{44} \\ \hline \end{array}$$

Главная диагональ

$$i = j$$

- 1) $i = j$ – элемент находится на главной диагонали;
- 2) $i + j = n + 1$ (n – размерность матрицы) – элемент находится на второй главной диагонали;
- 3) $i < j$ – элемент находится над главной диагональю;
- 4) $i > j$ – элемент находится под главной диагональю.

Элементы главной диагонали у матриц A и B одни и те же. Следовательно операция транспонирования матрицы A сводится к перестановке строк матрицы.

▣ *Транспонированной матрицей $B(n, n)$*

называется такая квадратная матрица, у которой столбцы соответствуют строкам исходной квадратной матрицы $A(n, n)$.

▣ Элементы главной диагонали у матриц A и B одни и те же. Следовательно операция транспонирования матрицы A сводится к перестановке строк матрицы.

$$\mathbf{A} = \begin{pmatrix} \mathbf{1} & \mathbf{2} & \mathbf{3} \\ \mathbf{4} & \mathbf{5} & \mathbf{6} \\ \mathbf{7} & \mathbf{8} & \mathbf{9} \end{pmatrix}; \quad \mathbf{B} = \begin{pmatrix} \mathbf{1} & \mathbf{4} & \mathbf{7} \\ \mathbf{2} & \mathbf{5} & \mathbf{8} \\ \mathbf{3} & \mathbf{6} & \mathbf{9} \end{pmatrix}$$

- ▣ **Sub** task_1()
- ▣ **Dim** a() **As Single**
- ▣ **Dim** b() **As Single**
- ▣ **n = 5**
- ▣ **ReDim** a(n, n): **ReDim** b(n, n)
- ▣ **For** i = 1 **To** n
 - ▣ **For** j = 1 **To** n
 - ▣ a(i, j) = **Cells**(i, j)
 - ▣ **b(j, i) = a(i, j)** 'транспонирование матрицы
 - ▣ **Next** j
- ▣ **Next** i
- ▣ **For** i = 1 **To** n
 - ▣ **For** j = 1 **To** n
 - ▣ **Cells**(n + 1 + i, j) = b(i, j) 'вывод результата
 - ▣ **Next** j
- ▣ **Next** i
- ▣ **End Sub**

	1	2	3	4	5	6
1	1	0,833333	0,714286	0,625	0,555556	
2	2	1,666667	1,428571	1,25	1,111111	
3	3	2,5	2,142857	1,875	1,666667	
4	4	3,333333	2,857143	2,5	2,222222	
5	5	4,166667	3,571429	3,125	2,777778	
6						
7	1	2	3	4	5	
8	0,833333	1,666667	2,5	3,333333	4,166667	
9	0,714286	1,428571	2,142857	2,857143	3,571429	
10	0,625	1,25	1,875	2,5	3,125	
11	0,555556	1,111111	1,666667	2,222222	2,777778	
12						