

## ЛАБОРАТОРНАЯ РАБОТА №5

### ЦИКЛ СО СЧЕТЧИКОМ. РАБОТА С МАССИВАМИ

#### *Цикл со счетчиком*

Такой цикл используется, если количество повторений известно.

Синтаксис:

**For** *счетчик* = начальное значение **To** конечное значение [**Step** шаг]

*Операторы*

[**Exit For**]

**Next** [*счетчик*]

Счетчик может быть как целым, так и действительным числом.

Шаг, равный 1, можно не указывать.

Шаг может быть как положительным, так и отрицательным, как целым, так и действительным числом.

Пример

*'Вычисление 10!*

**F=1**

**For I = 1 To 10**

**F=F\*I**

**Next I**

На рисунке 2.1 приведен фрагмент блок-схемы, соответствующий данному примеру.

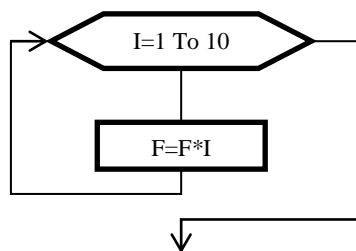


Рисунок 2.1

Цикл со счетчиком нередко используется для работы с массивами. Массив – это упорядоченное множество элементов одного типа, доступ к каждому элементу которого возможен по его индексу (номеру). В VBA возможны массивы фиксированные (с неизменным количеством элементов) и динамические (количество элементов может изменяться по ходу программы).

В одномерном массиве используется один индекс для доступа к элементам массива. Такой массив можно представить как строку или столбец с определенными значениями.

В двумерном массиве используются два индекса: номер строки и номер столбца, а сам массив представляет из себя таблицу с данными указанного при объявлении массива типа.

По умолчанию нумерация элементов начинается с нуля. В этом случае при объявлении массива указывается **номер** последнего элемента, а не количество элементов! Примеры объявления массивов:

**Dim M1(9) as Byte** *‘массив из 10 целых чисел*

**Dim M2(4,9) as Double** *‘двумерный массив 5x10 действит. чисел*

При объявлении массива начальный индекс можно задать:

**Dim M3(1 To 10) as Integer** *‘массив из 10 целых чисел*

**Dim M4(1 To 5,1 To 10) as String** *‘двумерный массив 5x10  
‘текстовых строк.*

#### ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ

Задача 2.1. Составить программу для нахождения всех нечетных чисел, расположенных в интервале [a, b], вычисления их суммы и количества. Исходные данные (числа a и b) получить из ячеек на листе MS Excel, нечетные числа вывести на лист, а их сумму и количество – в диалоговое окно и на лист. Проверить работу программы при a=150, b=199.

Решение

Пример программы с комментариями приведен ниже.

```

Public Sub Primer2_1()
Dim k As Integer, N As Integer, a As Integer, b As Integer
Dim Sum As Long
a = Range("A2") 'чтение начала отрезка
b = Range("B2") 'чтение конца отрезка
k = 0 'счетчик найденных чисел
Sum = 0 'сумма найденных чисел
For N = a To b
  If (N Mod 2) <> 0 Then 'Если число нечетное
    Sum = Sum + N 'добавляем число к сумме
    k = k + 1 'увеличиваем счетчик чисел
    Cells(k + 1, 3) = N 'выводим число в ячейку
  End If
Next N
Range("D2") = Sum 'вывод суммы в ячейку
Range("E2") = k 'вывод количества чисел в ячейку
'вывод в диалоговое окно
MsgBox "Сумма=" & Str(Sum) & vbNewLine & "Найдено" & _
Str(k) & " чисел"
End Sub

```

В этой программе для отбора нечетных чисел используется проверка остатка от деления числа на 2 (деление по модулю 2). Если он ненулевой, то число добавляется к сумме, счетчик найденных чисел увеличивается на 1, и число выводится в ячейку на лист. Номер строки ячейки зависит от счетчика чисел, поэтому каждое новое число выводится в следующей строке.

При формировании строки вывода в диалоговое окно (оператор `MsgBox`) числовые данные преобразуются к строковому типу функцией `Str`, отдельные элементы строки соединяются оператором `&` (конкатенация), для перехода на новую строку в диалоговом окне в месте перехода вставляется системная переменная `vbNewLine`.

Эту же задачу можно было решить по-другому. Перед выполнением цикла проверить, не является ли число *a* нечетным, если да, то цикл начать с него, в противном случае увеличить число

на 1 и начать цикл с него. А сам цикл выполнять с шагом 2, то есть рассматривать только нечетные числа. Пример соответствующей программы приведен ниже.

```
Public Sub Primer2_1_1()  
Dim k As Integer, N As Integer, a As Integer, b As Integer  
Dim Sum As Long  
a = Range("A2") 'чтение начала отрезка  
b = Range("B2") 'чтение конца отрезка  
k = 0 'счетчик найденных чисел  
Sum = 0 'сумма найденных чисел  
If (a Mod 2) = 0 Then a = a + 1 'Если число четное  
For N = a To b Step 2  
    Sum = Sum + N 'добавляем число к сумме  
    k = k + 1 'увеличиваем счетчик чисел  
    Cells(k + 1, 3) = N 'выводим число в ячейку  
Next N  
Range("D2") = Sum 'вывод суммы в ячейку  
Range("E2") = k 'вывод количества чисел в ячейку  
'вывод в диалоговое окно  
MsgBox "Сумма=" & Str(Sum) & vbCrLf & "Найдено" & _  
Str(k) & " чисел"  
End Sub
```

На рисунке 2.2 показан фрагмент листа с результатами (часть ячеек не показана).

	A	B	C	D	E	F
1	a	b	N	Сумма	Кол-во	
2	150	199	151	4375	25	
3			153			
4			155			
23			193			
24			195			
25			197			
26			199			
27						

Рисунок 2.2

Задача 2.2. Сформировать матрицу 10x10, элементы которой вычисляются в соответствии со следующим выражением:

$$A_{i,j} = \begin{cases} i + j, & \text{если } i = j \\ 2i + j, & \text{если } i > j \\ i - 2j, & \text{если } i < j \end{cases}$$

Значения  $i$ ,  $j$  и полученную матрицу вывести на лист, начиная с выделенной в момент запуска программы ячейки. Ниже, отступив одну пустую строку, вывести транспонированную матрицу, еще ниже через пустую строку – сумму первых двух матриц.

Решение

Несмотря на то, что размер матриц нам известен, лучше программу составить таким образом, чтобы конкретный размер (в нашем случае число 10) использовался в программе как можно реже. Для этого в начале программы объявим константу, равную заданному размеру и в тексте везде, где нужно будет указать размер, будем указывать не число 10, а имя нашей константы. Благодаря такому приему для формирования матриц другого размера

достаточно будет в одном месте изменить значение константы, вместо того, чтобы выискивать и изменять значение размера по всей программе.

Прежде чем формировать матрицы, необходимо запомнить текущую ячейку, так как вывод всех матриц будет осуществляться относительно нее. Для этого будем использовать переменную типа Object.

Для улучшения читаемости результатов изменим цвет всех заголовков на красный. Для этого будем изменять свойство .Font.Color соответствующей ячейки на красный, используя функцию RGB. Функция возвращает в программу код цвета в цветовой модели RGB на основании значений трех аргументов. Каждый аргумент может принимать целое значение от 0 до 255. Первый по порядку аргумент задает количество (яркость) красного цвета, второй – зеленого, третий – голубого. Например, если записать RGB(255,0,0), функция выдаст код красного цвета максимальной яркости, RGB(0,255,0) – зеленого, RGB(0,0,255) – голубого, RGB(0,0,0) – черного, RGB(255,255,255) – белого. Возможны любые комбинации яркости базовых цветов в диапазоне от 0 до 255.

Пример текста программы с необходимыми комментариями приведен ниже.

**Public Sub Primer2\_2()**

**Const R = 10** *'размер массива как константа*

**Dim M1(1 To R, 1 To R) As Integer, M2(1 To R, 1 To R) As Integer, M3(1 To R, 1 To R) As Integer**

**Dim i As Byte, j As Byte, Sred As Single**

**Dim AC As Object** *'для запоминания активной ячейки*

**Set AC = ActiveCell** *'активная ячейка*

*'Вывод на лист значений i и j*

**AC = "j \ i"**

*'изменение цвета шрифта в ячейке на красный*

**AC.Font.Color = RGB(255, 0, 0)**

**For i = 1 To R** *'Вывод номеров столбцов в строку*

**Cells(AC.Row, AC.Column + i) = i**

```

'изменение цвета шрифта в ячейке на красный
Cells(AC.Row, AC.Column + i).Font.Color = RGB(255, 0, 0)
Next i
For i = 1 To R 'Вывод номеров строк в столбец
Cells(AC.Row + i, AC.Column) = i
'изменение цвета шрифта в ячейке на красный
Cells(AC.Row + i, AC.Column).Font.Color = RGB(255, 0, 0)
Next i
'формирование первой матрицы
For i = 1 To R
For j = 1 To R
If i = j Then M1(i, j) = i + j
If i < j Then M1(i, j) = 2 * i + j
If i > j Then M1(i, j) = i - 2 * j
'вывод очередного элемента на лист
Cells(AC.Row + i, AC.Column + j) = M1(i, j)
Next j
Next i
'формирование и вывод транспонированной матрицы
Cells(AC.Row+R+1,AC.Column)="Транспонированная матрица"
'изменение цвета шрифта в ячейке на красный
Cells(AC.Row + R + 1, AC.Column).Font.Color = RGB(255, 0, 0)
For i = 1 To R
For j = 1 To R
M2(i, j) = M1(j, i)
Cells(AC.Row + R + i + 1, AC.Column + j) = M2(i, j)
Next j
Next i
'формирование и вывод суммы матриц
Cells(AC.Row + 2 * R + 2, AC.Column) = "Сумма матриц"
'изменение цвета шрифта в ячейке на красный
Cells(AC.Row + 2 * R + 2, AC.Column).Font.Color = RGB(255, 0, 0)
For i = 1 To R
For j = 1 To R
M3(i, j) = M1(i, j) + M2(i, j)
Cells(AC.Row + 2 * R + i + 2, AC.Column + j) = M3(i, j)

```

Next j  
 Next i  
 End Sub

Фрагмент листа с результатами работы программы показан на рисунке 2.3. Перед запуском программы была выделена ячейка D6, поэтому вывод начался с нее. Часть строк для сокращения размера рисунка не показана.

	D	E	F	G	H	I	J	K	L	M	N
6	j \ i	1	2	3	4	5	6	7	8	9	10
7		1	2	4	5	6	7	8	9	10	11
8		2	0	4	7	8	9	10	11	12	13
15		9	7	5	3	1	-1	-3	-5	-7	18
16		10	8	6	4	2	0	-2	-4	-6	-8
17	Транспонированная матрица										
18		2	0	1	2	3	4	5	6	7	8
19		4	4	-1	0	1	2	3	4	5	6
26		11	13	15	17	19	21	23	25	18	-8
27		12	14	16	18	20	22	24	26	28	20
28	Сумма матриц										
29		4	4	6	8	10	12	14	16	18	20
30		4	8	6	8	10	12	14	16	18	20
37		18	18	18	18	18	18	18	18	36	20
38		20	20	20	20	20	20	20	20	20	40

Рисунок 2.3

### ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Задача 2.1. Составить программу для нахождения всех чисел, делящихся без остатка на число  $c$ , расположенных в интервале  $[a, b]$ , вычисления их суммы, количества и среднего арифметического. Исходные данные (числа  $a$  и  $b$ ) получить из ячеек на листе MS Excel, найденные числа вывести на лист, а их сумму, количество и среднее арифметическое – в диалоговое окно и на лист.

Вариант	a	b	c
1	50	180	7
2	600	800	13



3	148	250	14
4	312	456	17
5	224	442	23
6	800	1000	29
7	720	930	15
8	860	1200	43
9	789	952	31
10	325	486	19

Задача 2.2. Сформировать матрицу  $R \times R$ , элементы которой вычисляются в соответствии с заданным выражением.

Значения  $i$ ,  $j$  и полученную матрицу вывести на лист, начиная с выделенной в момент запуска программы ячейки. Ниже, отступив одну пустую строку, вывести транспонированную матрицу, еще ниже через пустую строку – сумму первых двух матриц, еще ниже через пустую строку – разность первых двух матриц.

Вариант	R	Выражение
1	9	$A_{i,j} = \begin{cases} i * j, \text{если } i = j \\ i + j, \text{если } i > j \\ i - j, \text{если } i < j \end{cases}$
2	11	$A_{i,j} = \begin{cases} 1, \text{если } i = j \\ i + 2j, \text{если } i > j \\ i - j, \text{если } i < j \end{cases}$
3	13	$A_{i,j} = \begin{cases} -i, \text{если } i = j \\ 2(i + j), \text{если } i > j \\ i - j, \text{если } i < j \end{cases}$

4	12	$A_{i,j} = \begin{cases} i, \text{если } i = j \\  i - 2j , \text{если } i > j \\ 2i + 2, \text{если } i < j \end{cases}$
5	15	$A_{i,j} = \begin{cases} 2j, \text{если } i = j \\ i - j, \text{если } i > j \\ i + 2j, \text{если } i < j \end{cases}$
6	17	$A_{i,j} = \begin{cases} i * j, \text{если } i = j \\ i \bmod j, \text{если } i > j \\ i + j, \text{если } i < j \end{cases}$
7	14	$A_{i,j} = \begin{cases} i + j, \text{если } i = j \\ i \setminus j, \text{если } i > j \\ 3i + j, \text{если } i < j \end{cases}$
8	10	$A_{i,j} = \begin{cases} 1, \text{если } i = j \\ 2i \bmod j, \text{если } i > j \\ 3i - j, \text{если } i < j \end{cases}$
9	11	$A_{i,j} = \begin{cases} 3i, \text{если } i = j \\ i^2 \setminus j, \text{если } i > j \\ i - j, \text{если } i < j \end{cases}$
10	12	$A_{i,j} = \begin{cases} i - j, \text{если } i = j \\ i^2 \bmod j^2, \text{если } i > j \\ i^2 - j^2, \text{если } i < j \end{cases}$