

Оглавление

Оглавление	1
Переменные. Именованье	1
Массивы	1
Инструкции VBA	2
Оператор присвоения	2
Перенос строки	2
Комментарии	3
Расположение нескольких операторов на одной строке	3
Операции	3
Операции отношения	3
Условие IF	4
Логические операции	4
Другие операции	4
Математические функции	4
Функции преобразования форматов	5
Функции обработки строк	5
Цикл	6
Взаимодействие с Excel	7
Создание функции (Function)	7
Создание процедуры (Sub)	8
Пример задач	10

Переменные. Именованье

В VBA пользователь определяет имена переменных, функций, процедур, типов, постоянных и других объектов. Вводимые пользователем имена должны отражать суть обозначаемого объекта так, чтобы делать программу легко читаемой. В VBA имеются следующие ограничения на имена:

1. Длина имени не должна превышать 255 символов.
2. Имя не может содержать точек, пробелов и следующих символов: %, .&, !, #, @, \$.
3. Имя может содержать любую комбинацию букв, цифр и символов, начинающуюся с буквы (допускается использование русских букв, однако могут возникать проблемы при переносе на нерусифицированные компьютеры).
4. Имена должны быть уникальны внутри области, в которой они определены.
5. Не следует использовать имена, совпадающие с ключевыми словами VBA и именами встроенных функций и процедур.
6. Регистр букв (верхний или нижний) в имени не имеет значения, но умелое его использование может облегчить понимание содержания переменной.

Основные типы переменных

- Строковые (имеют окончание \$), например: `string$`
- Числовые, например: `numberOne`

Массивы

Объявление массива происходит следующим образом:

Visual Basic for Applications (VBA) для Excel. Быстрый старт для экономиста

`Dim A (12)` – объявляет одномерный массив (вектор) из 12 чисел, причем по умолчанию первый элемент массива будет `A (0)`, а последний `A (11)`.

`Dim A (1 To 12)` – объявляет массив (вектор) из 12 чисел от `A (1)` до `A (12)`.

Иногда в процессе выполнения программы требуется изменять размер массива. В этом случае первоначально массив объявляют как динамический. Для этого при объявлении массива не надо указывать размерность, например:

`Dim R ()`

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 R(n)`

Допустимо повторное использование инструкции `ReDim` для изменения числа элементов и размерностей массива.

Инструкции VBA

Инструкция VBA представляет собой полную команду языка VBA. Она может содержать ключевые слова, операторы, переменные, константы и выражения. В VBA имеются следующих три категории инструкций:

- Инструкции описания
- Инструкции присвоения
- Исполняемые инструкции

Оператор присвоения

Оператор присвоения присваивает значение выражения переменной, константе или свойству объекта. Оператор присвоения всегда включает знак равенства (=).

Переменная (или Постоянная или Свойство Объекта) = Выражение

Оператор присвоения предписывает выполнить выражение, заданное в его правой части, и присвоить результат переменной, имя которой указано в левой части. В результате выполнения следующего блока операторов

`x = 2`

`x = x + 2`

переменной `x` будет присвоено 4.

Перенос строки

Расположение символов (пробел) + (Знак подчеркивания) в конце строки обеспечивает то, что последующая строка является продолжением предыдущей. При этом надо помнить, что:

- нельзя разбивать переносом строковые константы;
- допустимо не более семи продолжений одной и той же строки;
- сама строка не может состоять более чем из 1024 символов.

`c = +_`
`Int(x)`

Комментарии

Работая с программой, удобно использовать комментарии, т. е. фрагменты текста программы, не являющиеся программными кодами и игнорируемые компилятором. Комментарии выполняют две важные функции:

1. Делают программу легко читаемой, поясняя смысл программных кодов и алгоритма.
2. Временно отключают фрагменты программы при ее отладке. В языке VBA существуют два способа ввода комментариев:
 - Применение апострофа ' . Его можно ставить в любом месте строки. При этом все символы, начиная от апострофа до конца строки, будут восприниматься компилятором как комментарий.
 - Применение зарезервированного слова Rem вместо апострофа.

Ниже приведен пример использования комментариев в тексте программы:

```
Dim students$(20)
' students$ – массив строковых переменных
Rem от students$(0) до students$(19)
```

Расположение нескольких операторов на одной строке

Использование знака двоеточия позволяет разместить несколько операторов на одной строке. Таким образом, следующие две конструкции эквивалентны:

$$\begin{aligned}x &= x + 1 & x &= x + 1 : y = x + 2 \\ y &= x + 2\end{aligned}$$

Операции

A=3 : B=5

A+B	Сложение	8
A - B	Вычитание	-2
- A	Перемена знака	-3
A * B	Умножение	15
A/ B	Деление	0,6
A \ B	Целочисленное деление	0
A Mod B	Остаток от деления по модулю	3
A ^ B	Возведение в степень	243

Операции отношения

A < B	Меньше	True
A > B	Больше	False
A <= B	Меньше или равно	True
A >= B	Больше или равно	False
A <> B	Не равно	True
A = B	Равно	False

Условие IF

Синтаксис:

```
If выражение Then
    инструкции
[Elseif выражениеДругое Then
    [инструкции_elseif]]
[Else
    [инструкции_elseif ]],
End If
```

Пример:

```
test = True Or False
If test = True Then
    result$ = "Правда"
Elseif a = False Then
    result$ = "Ложь"
Else
    result$ = "Никогда не будет присвоено"
End If
```

По результатам выполнения этого кода result\$ будет равно "Правда"

Логические операции

Используются в конструкции If в качестве Выражения

A1=A>B : B1=B>A

A1 And B1	Логическое умножение	False
A1 Or B1	Логическое сложение	True
A1 Xor B1	Исключающее or (или)	True
Not A1	Логическое отрицание	True

Другие операции

"сдать" & "зачет"	Сложение строк. Для сложения строк допустимо использование операции "сдать" + "зачет", но предпочтительнее, во избежания путаницы, применять операцию со знаком &	"сдатьзачет"
-------------------	---	--------------

Математические функции

Abs (A)	Модуль (абсолютная величина)числа	3
Cos (A/3)	Косинус	0,540302
Sin (A/3)	Синус	0,841471
Exp (A)	Экспонента, т. е. результат	20,08554

Visual Basic for Applications (VBA) для Excel. Быстрый старт для экономиста

	возведения основания натурального логарифма в указанную степень	
Log (A)	Натуральный логарифм	1,098612289
Rnd ()	Случайное число из интервала [0,1). Перед вызовом функции Rnd используйте инструкцию Randomize() без аргумента	☺
Sgn (A) Sgn (-A)	Знак числа	1 -1
Sqr (A)	Квадратный корень из числа	1,732050808
Int (A/B)	Отбрасывает дробную часть	0

Функции преобразования форматов

Val ("3.5")	Возвращает числа, содержащиеся в строке, как числовое значение соответствующего типа	3.5
Str (A)	Возвращает значение, являющееся строковым представлением числа	"3"

В качестве допустимого десятичного разделителя функция str воспринимает только точку. При наличии другого десятичного разделителя (например, запятой) для преобразования чисел в строки следует использовать функцию cstr, описанную ниже.

Функции обработки строк

string\$="сдать Зачет" : length=5 : start=4

Lease(string\$)	Преобразует строку к нижнему регистру.	"сдать зачет"
Ucase(string\$)	Преобразует строку к верхнему регистру.	"СДАТЬ ЗАЧЕТ"
Left(string\$, length)	Возвращает подстроку, состоящую из заданного числа первых символов исходной строки	"сдать"
Right(string\$, length)	Возвращает строку, состоящую из заданного числа последних символов исходной строки.	"зачет"
Mid(string\$, start, length)	Возвращает подстроку строки, содержащую указанное число символов.	"ть За"
Len(string\$)	Возвращает число символов строки.	11
LTrim("&string\$") RTrim(string\$&" ")	Возвращает копию строки без пробелов в начале конце.	"сдать Зачет" "сдать Зачет"
InStr(start, string\$, "a")	Возвращает позицию первого вхождения одной строки внутри другой строки.	8

Цикл

Повторяет блок операторов заданное количество раз.

```
FOR Счетчик = Начало TO Конец [STEP шаг]
  [Инструкции]
NEXT Счетчик
```

Пример:

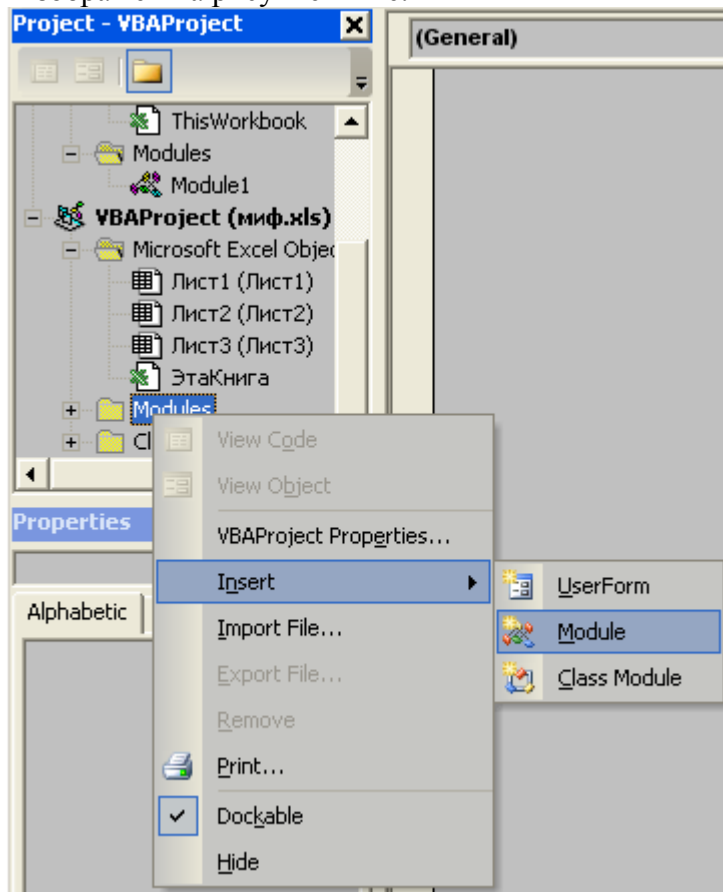
```
j = 0
For i = 1 To 10 step 1
  j = j + 1
Next i
String$ = Str$(i) & " " & Str$(j)
```

После выполнения данной программы String\$ принимает значение "11 10".

Взаимодействие с Excel

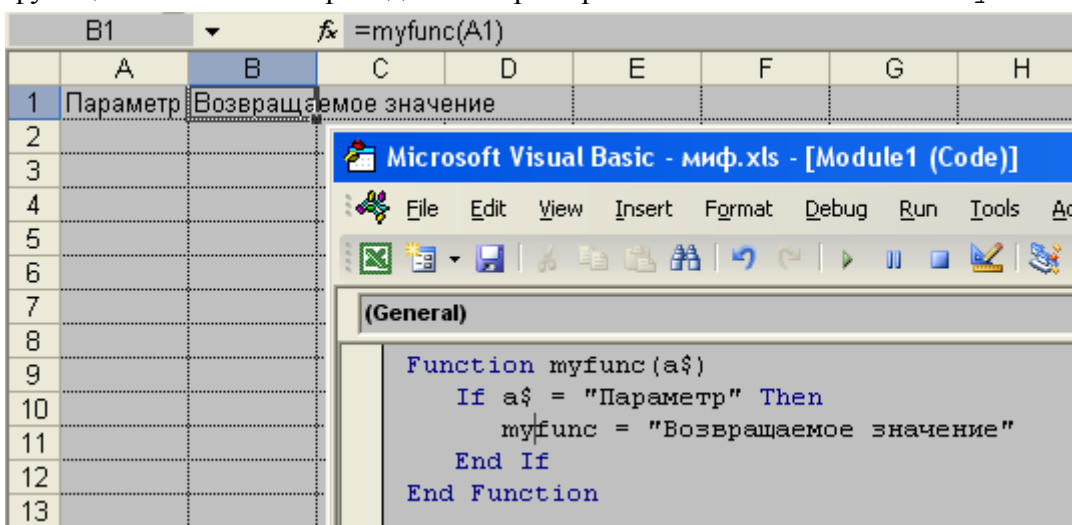
Код программ на VBA в Excel создается в специальном редакторе, который доступен по горячим клавишам Alt+F11 или через меню Сервис → Макрос → Редактор Visual Basic

Код программ записывается в специальных модулях. Пример создания нового модуля изображен на рисунке ниже.



Создание функции (Function)

Функция, созданная в VBA может быть вызвана как любая другая встроенная функция MS Excel. В приведенном примере в ячейке B1 вызывается myfunc, в



которую передается содержимое ячейки A1 и присваивается переменной a\$. Далее,

Visual Basic for Applications (VBA) для Excel. Быстрый старт для экономиста

в ходе выполнения программы одноименной с названием функции myfunc переменной присваивается значение, возвращаемое в рабочий лист Excel.

Создание процедуры (Sub)

Процедура (или подпрограмма или макрос) отличается от функции тем, что

1. в нее не передаются значения;
2. из нее не возвращаются значения;

⇒ процедура *не* может быть вызвана из ячейки листа.

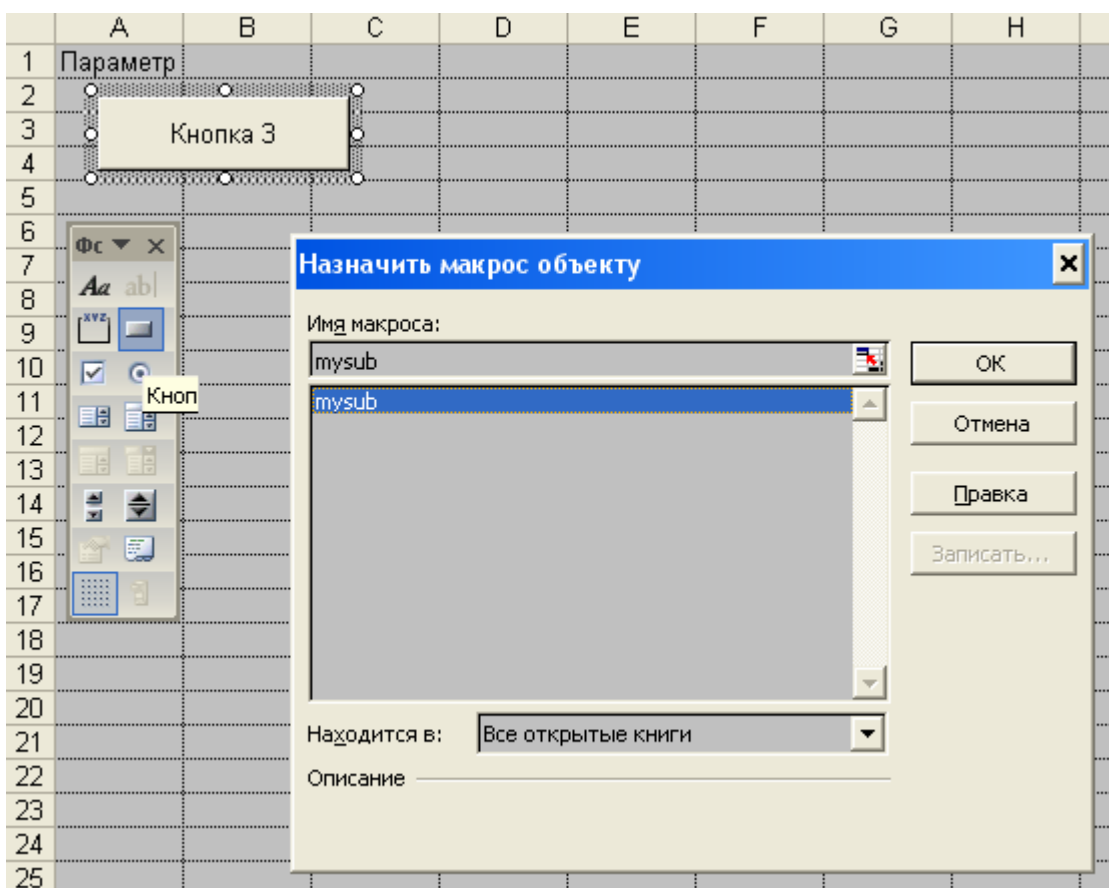
Процедуры должны быть полностью автономны и получать входные данные для расчетов из ячеек листа или из других источников самостоятельно.

Подпрограмма объявляется аналогично объявлению функции:

```
Sub mysub ()  
    Rem Операторы процедуры  
End Sub
```

Для запуска выполнения процедуры необходимо создать кнопку ее вызова.

Для этого нужно выбрать одноименную пиктограмму на панели инструментов Формы (Вид → Панели инструментов → Формы) и присвоить ей выполнение макроса. В случае, если процедура на момент создания кнопки еще не написана, то можно указать ее в поле «Имя макроса» и нажать на кнопку «Создать» (на рисунке внизу кнопка «Создать» заменит кнопку «Правка»).



В VBA предусмотрено несколько способов считывания данных с листа. Во следующих двух вариантах значение из ячейки A1 будет считываться в переменную a\$.

```
a$=Cells(1, 1).Value  
a$=Range("A1").Value
```


Visual Basic for Applications (VBA) для Excel. Быстрый старт для экономиста

При этом первый вариант удобен для считывания большого массива ячеек с использованием циклов. Второй – при считывании отдельных ячеек из листа.

The screenshot displays an Excel spreadsheet and the Microsoft Visual Basic editor. In the spreadsheet, cell A1 contains the text "Параметр" and cell B1 contains "Возвращаемое значение". A button labeled "Кнопка 3" is positioned over cells B3 and C4. The VBA editor window, titled "Microsoft Visual Basic - миф.xls - [Module2 (Code)]", shows the following code in the (General) tab:

```
Sub mysub ()  
  
    a$ = Cells(1, 1).Value  
    If a$ = "Параметр" Then  
        Cells(1, 2).Value = "Возвращаемое значение"  
    End If  
  
End Sub
```

На рисунке выше показан результат выполнения процедуры после нажатия кнопки «Кнопка 3» или кнопки «Run Sub/UserForm». Алгоритм работы процедуры аналогичен алгоритму работы функции рассмотренной в предыдущем разделе.

Пример задач

Задачи на технику и алгоритмизацию¹:

- 1) Найти корни квадратного уравнения при заданных параметрах a, b, c .²
- 2) Найти N -ное число Фибоначчи³.
- 3) Показать разложение числа на Общие Делители.
- 4) Найти Наибольший Общий Делитель (НОД) двух чисел.
- 5) Вывести в столбец все перестановки чисел $1..N$.
- 6) Подсчитать количество слов длины K из данных N букв, не содержащих данное подслово.
- 7) Привести матрицу к треугольному виду и решить систему уравнений методом Гаусса.

¹ Задачи пронумерованы по мере усложнения алгоритмической составляющей задачи.

² Нечетные задачи реализовать в виде подпрограмм, четные – в виде функций.

³ В том числе найти формулу определения чисел Фибоначчи.