

### Лекция 3

На предыдущей лекции мы рассмотрели вопросы создания и форматирования таблицы, в которую записаны значения трёх функций одного семейства.

После того, как таблица создана можно перейти к следующему шагу в создании документа, а именно к созданию и форматированию диаграммы – семейства графиков, обеспечивающего получение наиболее полного представления о поведении функций семейства.

Рассказывать, как выполняется построение и форматирование диаграммы я не буду. Вместо этого Вам предлагается файл (точнее 2 файла, упакованных в один). Этот упакованный файл "*Созд\_иФорматирДиаграмм.rar*" Вы сможете скопировать из папки "*Информатика 1-й курс*", находящейся в папке "*Темр*" на компьютерах 253 классов.

**Внимание !!!** При копировании файлов из папки "*Темр*" не следует их "*перетаскивать*" мышкой, т.к. при этом Вы воруете этот файл в свою папку, не оставляя его копии в папке "*Темр*". Копировать надо либо выполняя два шага: "*Копировать*" и "*Вставить*", либо, при перетаскивании файлов из окна в окно следует нажать клавишу *Ctrl*.

В упакованном файле "*Созд\_иФорматДиаграмм.rar*" находится файл методических указаний по созданию и форматированию диаграмм. На последней странице этого документа находится окончательно оформленный документ перед выводом его на печать.

В рассмотренном нами примере мы познакомились с методикой записи формул в ячейки таблицы программы *Excel*. Так для реализации формулы

$$F(x) = a \cdot \sin(b \cdot x)^{(1/3)} = a \cdot \sqrt[3]{\sin(b \cdot x)} \quad (1)$$

мы записали в ячейку таблицы формулу

$$=B\$2*sin(\$D\$2*A4)^(1/3) \quad (2)$$

На примере этой формулы мы познакомились с правилами создания абсолютных и относительных ссылок, с правилами записи арифметических выражений для выполнения операций, имеющих разные приоритеты и, наконец, с правилами записи обращений к функциям, включённым в набор инструментов программы *Excel*.

Рассмотрим эти вопросы более подробно.

### Основные правила записи выражений

В записи формульных алгоритмов обработки данных чаще всего используются формулы, реализующие арифметические и логические выражения.

### Построение формул

Структура и порядок операндов в формуле определяет конечный результат вычислений. Правила (синтаксису), задающие порядок записи формул в Microsoft Excel, определяют, что в формулу включается знак равенства (=), за которым идут вычисляемые элементы (операнды), разделенные операторами (знаками операций).

В качестве операндов могут выступать постоянные значения (константы), ссылки на ячейки или диапазоны ячеек, заголовки, имена или функции.

В Excel действия выполняются слева направо — с учетом приоритета операторов, начиная от знака равенства (=).

Порядком вычисления можно управлять, группируя действия с помощью скобок. Выражения, заключенные в скобки должны вычисляться в первую очередь. Например, результатом вычисления значения по формуле

$$=5+2*3$$

будет число 11, поскольку *Excel* выполняет умножение до сложения. В данной формуле число 2 умножается на 3, а затем к результату добавляется число 5.

Если же с помощью скобок изменить формулу, записав её как

$$=(5+2)*3$$

то результатом будет число 21. *Excel* сначала сложит числа 5 и 2, а затем умножит результат на 3.

### Порядок выполнения действий в формулах

В общем случае порядок выполнения действий в формулах в *Microsoft Excel* определяется приоритетом операторов. Вначале выполняются операции с более высоким приоритетом, затем — с менее высоким. Если формула содержит операторы с одинаковым приоритетом, например, операторы умножения и деления, то операции выполняются слева направо.

Оператор	Описание
'!' (двоеточие) и ',' (запятая)	Операторы ссылок.
-	Унарный минус (например -1).
%	Процент
^	Возведение в степень.
* и /	Умножение и деление.
+ и -	Сложение и вычитание.
&	Объединение двух текстовых строк в одну.
= < > <= >= <>	Операторы сравнения.

### Применение операторов в формулах

Операторами обозначаются операции - действия, которые следует выполнить над операндами формулы. В *Microsoft Excel* включено четыре вида операторов: арифметические, текстовые, операторы сравнения и операторы ссылок.

**Арифметические операторы.** Служат для выполнения арифметических операций, таких как сложение, вычитание, умножение. Операции выполняются над числами. Используются следующие арифметические операторы.

Арифметический оператор	Значение	Пример
+ (знак плюс)	Сложение	3+3
- (знак минус)	Вычитание Унарный минус	3-1 -1
* (звездочка)	Умножение	3*3
/ (косая черта)	Деление	3/3
% (знак процента)	Процент	20%
^ (крышка)	Возведение в степень	3^2 (аналогично 3*3)

**Операторы сравнения (отношения)** Используются для сравнения двух значений.

Результатом выполнения операции сравнения является логическое значение: либо ИСТИНА, либо ЛОЖЬ.

Операторы сравнения	Значение	Пример
= (знак равенства)	Равно	A1=B1
> (знак больше)	Больше	A1>B1
< (знак меньше)	Меньше	A1<B1
>= (знак больше и знак равенства)	Больше или равно	A1>=B1

<= (знак меньше и знак равенства)	Меньше или равно	A1<=B1
<> (знак «не равно»)	Не равно	A1<>B1

**Текстовый оператор конкатенации.** Амперсанд (&) используется для объединения нескольких текстовых строк в одну строку.

Текстовый оператор	Значение	Пример
& (амперсанд)	Объединение последовательностей символов в одну последовательность.	Выражение "Северный " & "ветер" эквивалентно строке "Северный ветер".

**Оператор ссылки.** Для описания ссылок на диапазоны ячеек используются следующие операторы.

Оператор ссылки	Значение	Пример
':' (двоеточие)	Оператор диапазона используется для ссылки на все ячейки между крайними ячейками диапазона, включая эти ячейки.	B5:B15
',' (запятая)	Оператор объединения объединяет в записи несколько ссылок в одну составную ссылку.	СУММ(B5:B15,D5:D15)

### Использование функций для вычисления значений

**Функции** — заранее определенные формулы, которые выполняют вычисления с заданными величинам, называемыми аргументами. Функции реализуют некоторые алгоритмы, ассоциированные с их названиями.

Например, функция **СУММ** суммирует значения в нескольких ячейках или нескольких диапазонах ячеек. Функция **SIN** вычисляет значение синуса для угла, определяемого значением выражения, использованного для записи **аргумента** функции.

**Аргументы функции** – это либо числовые константы, либо выражения, записанные с использованием констант и ссылок.

**Список аргументов** может содержать один или несколько аргументов, отделяемых друг от друга запятыми. Тип каждого из аргументов в списке определяется алгоритмом, реализуемым функцией. Это могут быть числа, тексты, логические величины (например ИСТИНА или ЛОЖЬ). Аргументами могут быть также выражения, результат вычисления которых соответствует требуемому типу операнда.

При записи аргументов в обращениях к функциям необходимо следить за соответствием их типов типам аргументов, входящих в определение функции.

### Создание пользовательских функций

Как уже отмечалось ранее в среде программы *Microsoft Excel* реализовано большое число функций, каждая из которых фактически заменяет некоторый часто и многократно используемый вычислительный алгоритм.

Обращение к стандартным функциям реализуется либо из списка 10 функций в строке формул, либо, как уже неоднократно отмечалось, через ДО "**Мастер функций – шаг 1 из 2**". Это окно открывается при выборе в списке строки формул варианта "**Другие функции...**".

Работая в среде программы *Excel*, пользователь может включить в файл своей книги собственные функции – функции пользователя. Для создания таких функций используются средства языка программирования *Visual Basic for Applications*.

Рассмотрим процесс создания функции пользователя на примере реализации функции (1). В *Microsoft Office* вводится такое понятие, как **МАКРОС**.

### Что такое макрос?

Если в каком-либо из приложений *Microsoft Office* необходимо многократно выполнять какую-то задачу, то лучше всего это сделать с помощью макроса.

**Макрос** — это серия команд, сгруппированных вместе для упрощения многократно выполняемой работы. Вместо того чтобы вручную делать отнимающие много времени и повторяющиеся действия, можно создать и запускать один макрос, который будет выполнять эту задачу. (В традиционном программировании такие последовательности чаще всего определялись как процедуры или функции.)

Макросы часто используются для реализации решения следующих задач:

- Для ускорения часто выполняемых операций редактирования или форматирования;
- Для объединения нескольких команд, например, для вставки таблицы с указанными размерами и границами и определенным числом строк и столбцов;
- Для упрощения доступа к параметрам в диалоговых окнах;
- Для автоматизации многократного выполнения последовательных действий в задачах.

### Создание функций пользователя как макросов VBA

Продолжим работу с уже имеющейся у нас таблицей. Создадим пользовательскую функцию, реализующую вычисление по формуле (1).

Рассмотрим последовательность действий, выполняемых при создании пользовательских функций и включении их в библиотеку функций программы *Excel*.

Для создания новой функции, включаемой в библиотеку функций программы *Excel*, следует:

1. В меню "Сервис" выбираем команду "Макрос ►" и далее выполняем команду "Редактор Visual Basic". Открывающееся при этом окно программы **Microsoft Visual Basic** показано на рис. 1.

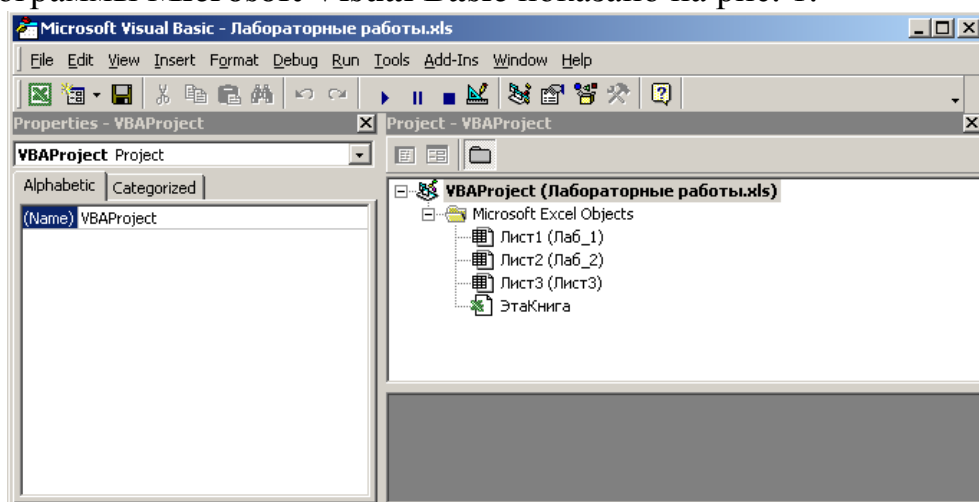


Рис. 1. Окно программы **Microsoft Visual Basic**

В этом окне открыты два рабочих окна: окно проекта – *Project* и окно *Properties* – свойства.

2. В окне программы **Microsoft Visual Basic** выбираем меню "**Insert**" (*Вставка*). В этом меню выбирается и выполняется команда **Module** (*Модуль*). В результате выполнения этой команды в проект рабочей книги включается папка "**Modules**" (*Модули*), в которой создается первый модуль - "**Module1**". рабочее которого открывается. Под полем проекта (**Project - VBAProject**) поле редактора, в котором будут создаваться пользовательские функции и процедуры.
3. В меню "**Insert**" (*Вставка*) окна **Microsoft Visual Basic** выбираем команду "**Procedure...**" (*Процедура*). Открывается окно "**Add Procedure**" (*Вставить процедуру*).

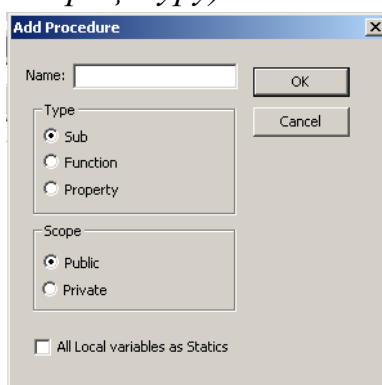


Рис. 2

В поле **Name** (*Имя*) вводим имя создаваемой функции. Например, **F** или **F1V**. **Очень важно**, чтобы имя функции не совпадало с именем какой-либо ячейки на листе книги. (Имена ячеек принимают значения от **A1** до **IV65536**). Нарушение этого условия приводит к появлению ошибок при выполнении процедур и функций.

4. В группе **Type** (*Тип*) выбираем переключатель, определяющий тип создаваемой процедуры. Тип **Sub** (*Подпрограмма*) может быть использован для создания процедур, выполняющих какие-либо действия и не передающих в основную программу числовые значения. Тип процедуры **Function** (*Функция*) используется для определения подпрограмм, результатом выполнения которых является одно числовое или логическое значение, передаваемое (возвращаемое) в основную программу через имя функции. В рассматриваемом примере выбираем переключатель типа **Function**.
5. В поле **Scope** (*Область действия*) выбираем переключатель **Public** (*Общедоступный*), обеспечивая тем самым доступность функции пользователя из любых других книг.
6. После нажатия кнопки **OK** в поле редактора вносятся две строки заготовки функции:

## Public Function F()

### End Function

7. Используя эту заготовку, допишем функцию пользователя соответствующую вычислениям по формуле (1).  $F=a*\sin(b*x)^{(1/3)}$ . Но такая запись не позволяет получить требуемый результат. Это объясняется тем, что в VBA не разрешена операция возведения отрицательного числа в дробную степень!!!. Поэтому запишем функцию следующим образом:

```
Public Function F(x, a, b)
    Q = Sin(b * x)
    If Q > 0 Then F = Q ^ (1 / 3) Else F = -(-Q) ^ (1 / 3)
    F = a * F
End Function
```

8. Для проверки работоспособности созданной функции **F**, используя "**Мастер функций**", запишем обращение к ней в ячейку **E5**. Для этого в окне первого шага "**Мастера функций**" в поле "**Категория:**" выбираем категорию "**Определенные пользователем**". Выбор функции **F** и нажатие кнопки ОК открывает ДО для определения параметров обращения. На рис. 3 показан результат определения ссылок в полях окна второго шага "Мастера функций"

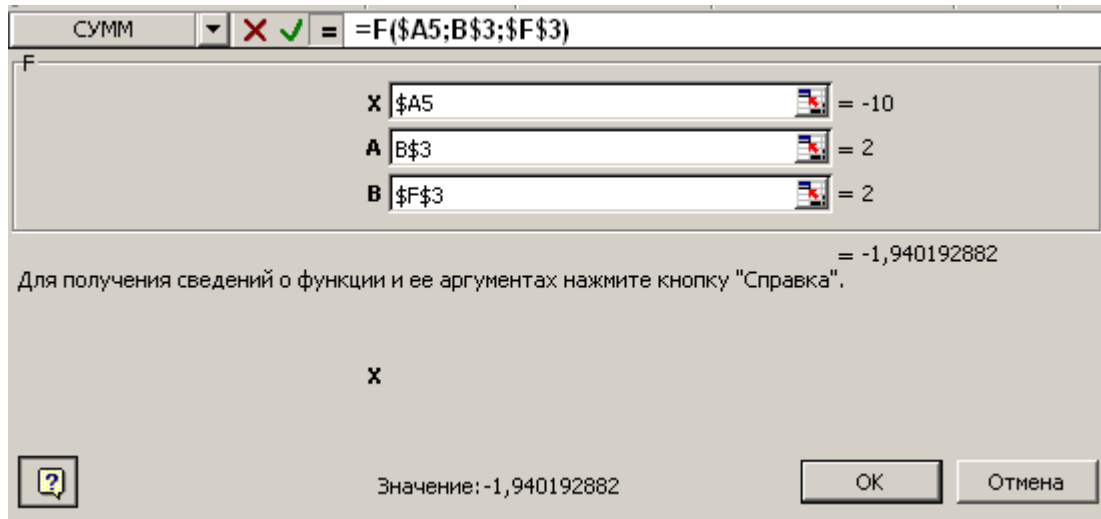


Рис. 3

Следует обратить внимание на то, что ссылки на параметры **A** и **B** должны быть абсолютными. Этим обеспечивается возможность "протягивания" формулы для вычисления значений функции **F** в ячейках от **E5** до **G25**.

Сравнивая значения функций в столбцах **B**, **C** и **D** со значениями полученными в столбцах **E**, **F** и **G**, мы можем убедиться в том, что реализованная нами функция работает правильно.