# МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Металлорежущие станки и инструменты»

# ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Методические рекомендации к управляемой самостоятельной работе для студентов специальности 1-40 05 01 «Информационные системы и технологии (по направлениям)» дневной формы обучения



Могилев 2022

### Рекомендовано к изданию учебно-методическим отделом Белорусско-Российского университета

Одобрено кафедрой «Металлорежущие станки и инструменты» «21» декабря 2021 г., протокол № 6

Составитель канд. техн. наук, доц. С. Н. Хатетовский

Рецензент канд. техн. наук, доц. М. Н. Миронова

Методические рекомендации к управляемой самостоятельной работе предназначены для студентов специальности 1-40 05 01 «Информационные системы и технологии (по направлениям)» дневной формы обучения.

Учебно-методическое издание

### ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Ответственный за выпуск	С. Н. Хатетовский
Корректор	И. В. Голубцова
Компьютерная верстка	М. М. Дударева

Подписано в печать 09.03.2022. Формат 60×84/16. Бумага офсетная. Гарнитура Таймс. Печать трафаретная. Усл. печ. л. 1,16. Уч.-изд. л. 1,27. Тираж 38 экз. Заказ № 69.

Издатель и полиграфическое исполнение: Межгосударственное образовательное учреждение высшего образования «Белорусско-Российский университет». Свидетельство о государственной регистрации издателя, изготовителя, распространителя печатных изданий № 1/156 от 07.03.2019. Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский университет, 2022

# Содержание

Введение	4
1 Тема № 1. Табличные процессоры	5
1.1 Краткие сведения о табличном процессоре Excel	5
1.2 Темы рефератов о табличных процессорах	7
2 Темы № 2 и 3. Программирование в среде табличного процессора. Ме-	-
тоды вычислений	8
2.1 Краткие сведения о методах вычислений и VBA	8
2.2 Темы рефератов о VBA	. 14
2.3 Темы рефератов о методах вычислений	. 14
3 Тема № 4. Математические программные пакеты	15
3.1 Краткие сведения о Mathcad	. 15
3.2 Темы рефератов о математических программных пакетах	. 19
Список литературы	. 19

# Введение

Целью выполнения управляемой самостоятельной работы является изучение обучающимися основ специализированного программного обеспечения.

Управляемая самостоятельная работа выполняется путем проработки тем (вопросов), вынесенных на самостоятельное изучение с конспектированием учебной литературы.

Обучающийся подготавливает реферат по предложенной преподавателем теме (допускается самостоятельный выбор темы, не противоречившей выбору остальных обучающихся группы).

Реферат оформляется в виде текстового документа объемом до 10 страниц в соответствии с ГОСТ 2.105.

При необходимости обучающийся может подготовить иллюстрирующие плакаты по рассматриваемой теме формата A1. Количество иллюстрационного материала ограничивается только необходимостью полного раскрытия темы реферата.

# 1 Тема № 1. Табличные процессоры

### 1.1 Краткие сведения о табличном процессоре Excel

Для представления данных в удобном виде используют таблицы.

Табличный процессор – это прикладная программа, которая предназначена для обработки данных, представленных в таблицах.

Таблицы, в которых представлены данные для обработки табличным процессором, называются электронными таблицами.

С ячейкой таблицы может быть связана определенная формула, в которую могут входить т. н. адреса – уникальные идентификаторы каких-либо других ячеек. При изменении значений в ячейках, адреса которых используются в формулах, изменяются значения ячеек, с которыми связаны указанные формулы.

Первый табличный процессор создали в 1979 г. студент Гарвардского университета (США) Дэн Бриклин и его товарищ, программист Боб Френкстон. Он был разработан для компьютера Apple II и получил название VisiCalc (англ. Visible Calculator – наглядный калькулятор). Программа обеспечивала возможность работать с одной таблицей, которая содержала 254 строки и 63 столбца, а также строить несложные диаграммы.

После VisiCalc появились аналогичные программы от других разработчиков и для других моделей компьютеров, такие как SuperCalc, Lotus 1-2-3, Multiplan и др. Впоследствии табличные процессоры начали включать в интегрированные пакеты прикладных программ, таких как Framework, Works и др.

Первая версия табличного процессора Excel появилась в 1985 г. Эта программа была создана для компьютера Apple и ее авторами были американские программисты Дуг Кландер и Филипп Флоренце. Первая версия Excel под OC Windows появилась в 1987 г.

Из других современных табличных процессоров можно назвать:

– Calc (англ. calculator – вычислитель, калькулятор), который входит в пакет программ StarOffice;

– GNumeric (англ. GNU – проект по созданию программного обеспечения свободного распространения, numeric – числовой), который свободно распространяется.

Интерфейс табличного процессора Excel, как и интерфейсы других составляющих MS Office, является классическим примером в OC MS Windows.

После запуска Excel на экране открываются два окна: окно табличного процессора и окно созданного в нем документа. Документ, создаваемый в табличном процессоре, называется книгой и по умолчанию получает имя «Книга1». Вновь созданная в Excel книга состоит из трех листов с именами «Лист1», «Лист2» и «Лист3».

На листах размещаются таблицы, но также могут быть размещены диаграммы, графики, графические изображения и др.

Каждая ячейка таблицы имеет адрес. Адрес ячейки указывается при работе

с формулами и диалогами Excel.

Адрес ячейки задается номерами столбца и строки, на пересечении которых она находится. Например, A1, C3, D17, AA26.

Чтобы выделить ячейку, достаточно по ней щелкнуть левой клавишей мышки. При этом ячейка становится текущей.

Всегда одна из ячеек таблицы текущая. Она обозначается табличным процессором черной рамкой (т. н. табличный курсор), а в поле имени в строке формул табличным процессором указывается ее адрес.

Столбцы электронной таблицы могут также нумероваться натуральными числами. В этом случае адреса ячеек указываются, например, так: R1C1, R5C2, R17C4, R27C26, где после буквы R (англ. Row – строка) указывается номер строки, а после С (англ. Column – столбец) – номер столбца. Чтобы установить числовой способ нумерации столбцов, необходимо выполнить настройку Excel (посредством кнопки «Office»).

Диапазон ячеек, как и отдельная ячейка, имеет адрес. Прямоугольный диапазон ячеек, который состоит из смежных ячеек, называется связным. Адрес связного диапазона ячеек представляет собой адреса двух ячеек, расположенных в его противоположных углах, которые разделены двоеточием. Например, A3:A7, B11:D11, G9:C3.

Строка и столбец также являются диапазонами ячеек. Например, адресом диапазона ячеек, в который входят все ячейки десятой строки, является адрес 10:10, а адресом диапазона ячеек, в который входят все ячейки столбца В, – адрес В:В. Соответственно, 6:8 – это адрес диапазона ячеек, который включает все ячейки строк с номерами 6, 7, 8, а H:L – адрес диапазона ячеек, в который входят все ячейки столбцов H, I, J, K, L.

Адрес несвязного диапазона ячеек представляет собой адреса его связных частей, разделенных точкой с запятой: A3:A7;G9:C3;B11:D11;G11;E13.

В Excel также используются т. н. трехмерные ссылки - ссылки на ячейки других листов той же или других книг. Для ссылок на ячейки другого листа той же книги нужно перед адресом ячейки указать имя листа с восклицательным знаком. Например, Лист2!А1 является ссылкой на ячейку А1 листа Лист2. Если необходимо записать в формуле ссылку на ячейку другой книги, и она открыта, то перед адресом ячейки необходимо записать имя файла книги в квадратных скобках восклицательным знаком. И имя листа с Например, [Смета.xlsx]Лист2!С4 является ссылкой на ячейку С4, которая находится на листе Лист2 в книге с именем Смета.xlsx, открытой в текущий момент времени работы с программой Excel. Если в имени файла книги или в имени листа есть пропуски, то такое имя нужно заключить в одинарные кавычки. Например, '[Итоги I семестра.xlsx]10 Б'!С15. Если нужная книга закрыта, то следует указать в одинарных кавычках путь к файлу, в котором записана книга, имя файла книги в квадратных скобках и имя листа, а за кавычками – восклицательный Например, знак И адрес ячейки. 'D:\Teмa3\[Смета.xlsx]Лист2'!С4.

Выделение связного диапазона ячеек осуществляется стандартным для ОС Windows способом при помощи перемещения мышки с нажатой левой клавишей.

Навигация по ячейкам осуществляется перемещением табличного курсора. Табличный курсор перемещается щелчком мышки по ячейке, клавиатурными клавишами со стрелками, клавиатурными клавишами «PgUp», «PgDn», «Ctrl»+«Home» (перемещение табличного курсора в ячейку A1), «Ctrl»+«End» (перемещение табличного курсора в крайнюю правую нижнюю ячейку, имеющую значение) и некоторыми другими способами.

Данные, хранимые в ячейках, могут иметь различный формат. Управление форматированием данных осуществляется по команде «Формат ячеек», которая для выделенных ячеек доступна в контекстном меню, активируемом щелчком правой клавиши мышки.

В ячейке могут храниться данные общего формата, числовые данные, текстовые данные и т. п.

Пользователь может также создать свой формат данных.

Для ввода данных в ячейку достаточно сделать необходимую ячейку текущей, набрать соответствующие данные на клавиатуре и нажать клавишу «Enter». Во время ввода данных надпись «Готово» в строке состояния изменяется на «Ввод», данные отображаются как в текущей ячейке, так и в поле формул. Ввести данные в ячейку можно также с использованием поля формул. Для этого достаточно сделать необходимую ячейку текущей, установить текстовый курсор в поле формул, набрать данные на клавиатуре и нажать клавишу «Enter». Если во время ввода данных нажать клавишу «Esc», то ввод данных будет отменен.

Одни и те же данные можно ввести одновременно в несколько различных ячеек одного листа. Для этого достаточно: выделить ячейки (не обязательно смежные); ввести с клавиатуры данные; нажать клавиши «Ctrl»+«Enter».

Одни и те же данные можно ввести одновременно в одноименные ячейки различных листов. Для этого достаточно: выделить ячейку одного листа; выделить ярлыки листов (при нажатой клавише «Ctrl»); ввести данные в ячейку; нажать клавишу «Enter».

### 1.2 Темы рефератов о табличных процессорах

1 Построение графических зависимостей в среде Excel.

2 Встроенные функции табличного процессора Excel.

3 Обеспечение безопасной работы в Excel.

4 Сводные таблицы Excel.

5 История развития электронных таблиц.

# 2 Темы № 2 и 3. Программирование в среде табличного процессора. Методы вычислений

#### 2.1 Краткие сведения о методах вычислений и VBA

Слово «алгоритм» появилось в IX в. и связано с именем математика Аль-Хорезми, который сформулировал правила выполнения четырех арифметических действий над многозначными числами. В настоящее время понятие алгоритма – одно из фундаментальных понятий науки информатики. С одной стороны, алгоритм является предметом изучения такой отрасли математики, как теория алгоритмов, с другой стороны, в информатике существует неформальное определение алгоритма.

Алгоритм – это точно определенная последовательность действий для некоторого исполнителя, выполняемых по строго определенным правилам и приводящих через некоторое количество шагов решению К задачи. Исполнитель алгоритмов определяет элементарные действия, из которых алгоритм. Отдельные действия, составляющие формируется алгоритм, называются операциями. При этом под операцией понимается как какое-то единичное действие, например сложение, так и группа взаимосвязанных действий. Основными особенностями любого алгоритма являются решение задачи в обобщенном виде и возможность выполнять действия по решению задачи для конкретных значений. Основным исполнителем несложных алгоритмов выступает человек. При решении сложных задач исполнителем является ЭВМ и составление алгоритма решения задачи – необходимый этап, дальнейшего детализирующий метод решения для программирования. Программа осуществляет еще более глубокую детализацию решения и его визуализацию. Свойства алгоритма:

– определенность – выполнив очередное действие, исполнитель должен точно знать, что ему делать дальше;

- дискретность – прежде чем выполнить определенное действие, надо выполнить предыдущее;

– массовость – по одному и тому же алгоритму решаются однотипные задачи и неоднократно;

– понятность – алгоритм строится для конкретного исполнителя и должен быть ему понятен;

– результативность – алгоритм всегда должен приводить к результату.

Можно сказать, что в процессе формального решения задачи ее решение сначала описывается на языке математики в виде системы формул, а затем на языке алгоритмов в виде некоторого процесса, в котором используются ранее определенные математические формулы и условия их выполнения.

Алгоритм, реализующий решение задачи, можно представить различными способами: с помощью графического или текстового описания, в виде таблицы значений. Графический способ представления алгоритмов имеет ряд преимуществ благодаря визуальности и явному отображению процесса решения задачи. Алгоритмы, представленные графическими средствами, получили название «визуальные алгоритмы». Текстовое описание алгоритма является достаточно компактным и может быть реализовано на абстрактном или реальном языке программирования в виде программы для ЭВМ. Таблицы значений представляют алгоритм неявно, как некоторое преобразование конкретных исходных данных в выходные. Таким образом, все три способа представления алгоритмов можно считать взаимодополняющими друг друга. На этапе проектирования алгоритмов наилучшим способом является графическое представление, на этапе проверки алгоритма – табличное описание, на этапе применения – текстовая запись в виде программы.

При проектировании визуальных алгоритмов используют основные условные обозначения – символы по ГОСТ 19.701-90 (ISO 5807–85).

Основные правила выполнения схем алгоритмов:

– символы в схеме должны быть расположены равномерно; следует придерживаться разумной длины соединений и минимального числа длинных линий;

– не должны изменяться углы и другие параметры, влияющие на соответствующую форму символов; символы должны быть, по возможности, одного размера;

– символы могут быть вычерчены в любой ориентации, но предпочтительной является горизонтальная ориентация;

– минимальное количество текста, необходимого для понимания функции данного символа, следует помешать внутри данного символа; текст должен записываться слева направо и сверху вниз независимо от направления потока;

– если объем текста, помещаемого внутри символа, превышает его размеры, следует использовать символ комментария;

– если использование символов комментария может запутать или разрушить ход схемы, текст следует помещать на отдельном листе и давать перекрестную ссылку на символ;

– в схемах может использоваться идентификатор символа, который определяет символ для использования в справочных целях в других элементах документации (например, в листинге программы); идентификатор символа должен располагаться слева над символом;

– потоки данных или потоки управления в схемах показываются линиями;
направление потока слева направо и сверху вниз считается стандартным;

– в случаях, когда необходимо внести большую ясность в схему (например, при соединениях), на линиях используются стрелки; если поток имеет направление, отличное от стандартного, стрелки должны указывать это направление;

- в схемах следует избегать пересечения линий;

– две или более входящие линии могут объединяться в одну исходящую линию; если две или более линии объединяются в одну линию, место объединения должно быть сметено;

 – линии в схемах должны подходить к символу либо слева, либо сверху, а исходить либо справа, либо снизу; линии должны быть направлены к центру символа; – при необходимости линии в схемах следует разрывать для избежания излишних пересечений или слишком длинных линий, а также, если схема состоит из нескольких страниц; соединитель в начале разрыва называется внешним соединителем, а соединитель в конце разрыва – внутренним соединителем.

Модуль – это именованная единица, состоящая из одной или нескольких подпрограмм, а также объявлений, относящихся ко всем этим подпрограммам. Хотя VBA и допускает размещение всех подпрограмм в одном модуле, имеет смысл разместить их в нескольких модулях в соответствии с выполняемыми этими подпрограммами задачами, чтобы с ними было проще работать. В VBA имеется два типа модулей. Чаще всего используется стандартный модуль, который содержит программный код, предназначенный непосредственно для выполнения. Другим типом модуля является модуль класса, в котором определяются пользовательские объекты с их свойствами и методами.

В модуле программа VBA обычно начинается с опций, которые управляют описанием переменных, способом сравнения строк и т. д.

Затем обычно следует объявление переменных или констант. Далее обычно располагаются непосредственно подпрограммы пользователя, составляющие саму программу. При написании программ удобно использовать комментарии.

Комментарии – это пояснительный текст, который можно записать в любом месте программы. Каждая строка комментариев начинается со знака апострофа. Комментарии игнорируются VBA и поэтому никакого влияния на программу не оказывают. Их удобно использовать также при отладке операторов для их временного отключения. Для более компактной записи программы используется символ двоеточия в качестве разделителя некоторых операторов в одной строке.

Проектом в VBA называется документ, включающий модули, формы и связанные с приложением объекты.

Переменные в VBA должны иметь имена (идентификаторы). Прежде чем начать работать с переменной, ей надо присвоить уникальное имя и объявить ее как переменную. Имя для переменной может быть выбрано произвольно, но с соблюдением следующих правил:

– максимальная длина имени переменной не может превышать 255 символов;

– имена переменных должны быть уникальными в их области действия;

– первым символом в имени переменной должна быть буква, за которой могут следовать в любой комбинации цифры, буквы или символ подчеркивания;

– имя переменной не может содержать пробелов, точек, восклицательных знаков, а также символов (a), #, \$, %, &;

– в качестве имени переменной нельзя использовать ключевые слова языка VBA.

Регистр символов в именах переменных не учитывается.

Все переменные в VBA имеют тип. Тип указывает, что может хранить переменная: целое или вещественное число, строку, дату и т. д.

В приложении VBA можно создавать следующие основные типы

подпрограмм:

– подпрограмма **Sub** (процедура, содержит набор команд, с помощью которого можно решить некоторую задачу);

– подпрограмма **Function** (функция, содержит набор команд, который решает некоторую задачу и при этом возвращает некоторое значение);

– подпрограмма **Property** (свойство, предназначена для ссылки на свойство объекта).

Аргументы процедуры – это данные, передаваемые процедуре при ее вызове и используемые ею при выполнении. При объявлении процедуры аргументы указываются в виде списка аргументов. Этот список заключается в скобки и записывается непосредственно после имени процедуры, причем отдельные элементы списка отделяются запятыми. Для каждого из аргументов указывается его имя и тип данных, записываемый после ключевого слова As. Тип аргумента может быть любым стандартным типом языка VBA или объявленным классом. Если указано только имя аргумента без указания его типа, то VBA приписывает такому аргументу тип Variant. В языке VBA процедура может иметь и переменное число аргументов. Для создания процедуры с переменным числом аргументов используется ключевое слово ParamArray, определяющее массив, который будет содержать необходимое количество значений-аргументов. Ключевое слово ParamArray указывается в объявлении процедуры только один раз и обязательно перед тем аргументом, который в списке стоит последним.

Передавать аргументы процедуре можно двумя способами – либо как ссылку, либо как значение. При передаче аргумента по ссылке (т. е. указывая адрес его размещения в памяти) процедура получает непосредственный доступ к этому аргументу. В результате при выполнении она может изменить значение переданного по ссылке аргумента.

При передаче аргумента по значению процедура получает копию передаваемого аргумента, и поэтому изменение аргумента внутри процедуры уже не оказывает влияния на исходную переменную.

Для организации передачи аргумента по значению используется ключевое слово **ByVal**, которое ставится перед именем этого аргумента в объявлении вызываемой функции. Способ передачи аргументов по ссылке принят в VBA по умолчанию. Если в качестве аргумента используется переменная, тип данных которой определен пользователем, то такой аргумент можно передавать только в виде ссылки.

Если процедура не имеет аргументов, то ее можно вызвать, просто указав имя этой процедуры.

Выражения позволяют получить определенный результат, имеющий один из допустимых в VBA типов. Выражения в VBA строятся из переменных, констант, функций, массивов, элементов массивов и операторов, выполняющих в определенной последовательности какие-либо операции. В простейшем случае выражение может состоять из одной переменой или константы или включать одну операцию.

Результат выражения присваивается переменной и становится ее

значением. Если переменной численного типа присваивается численный результат выражения, который выходит из диапазона значений типа переменной, VBA округляет результат выражения. Переменная типа Variant может иметь значение любого типа. Реальный тип этой переменной – тип ее значения. Если переменной типа String присваивается результат выражения типа Variant, представляющий собой число, VBA автоматически преобразует это число в строку. Переменная типа Variant может также содержать особые значения: Empty, Null, Error, Nothing. Значение Null означает отсутствие действительного значения. Значение Empty показывает, что переменная неинициализирована. В арифметических операциях это значение трактуется как 0, в логических – как False, в строковых операциях – как строка, не содержащая символов. Error означает, что в выражении возникла ошибка. Nothing показывает отсутствие ссылки на значение типа Object.

VBA может выполнять некоторые арифметические операции над числами: сложение, вычитание, умножение и деление, а также возведение в степень и предоставляет дополнительные математические операции: целочисленного деления и деления по модулю. Соответствующие операторы: +, -, \*, /, ^, \, Mod. В качестве каждого операнда может выступать переменная, константа, элемент массива, функция, результат другой операции. При этом каждый операнд должен иметь численный тип или тип String. В последнем случае VBA стремится автоматически преобразовать строку в численный тип. Если это невозможно, возникает ошибка. Результат операции имеет такой же тип, что и тип операнда с наибольшим диапазоном значений. Результат операции над операндом типа Single и операндом типа Long имеет тип Double. Если любой операнд в операции является равным Null, то результатом операции также будет Null.

Тип результата операции деления операндов целочисленных типов – **Double**. Если один из операндов операции деления имеет как наиболее широкий по диапазону значений тип **Single**, то результат также имеет этот тип, если этот результат не выходит из диапазона значений типа **Single**, в противном случае результат имеет тип **Double**.

Целочисленное деление и деление по модулю всегда имеют результатом целое число. При этом VBA округляет каждый операнд (если необходимо) до целого числа, тип результата целочисленного деления – либо Integer, либо Long: VBA использует наименьший подходящий тип данных для результата.

Операции сравнения иногда также называют операциями отношения. Результатом любой операции сравнения является значение типа Boolean: True или False. Операции сравнения (>, >=, <, <=, =) используются для сравнения значений.

Приоритеты выполнения операций:

– части выражения, заключенные в круглые скобки, всегда вычисляются в первую очередь; если выражение, заключенное в круглые скобки, является другим сложным выражением, VBA применяет эти же правила к выражению в круглых скобках;

– конкретные операции выполняются в зависимости от иерархии операторов;

 когда операторы имеют равный уровень приоритета, они вычисляются в порядке слева направо.

VBA вычисляет операторы в следующем порядке:

- арифметические операторы;

- оператор конкатенации строк;

– операторы сравнения;

– логические операторы.

Интерпретатор языка VBA выполняет операторы последовательно. Однако в некоторых случаях необходимо управлять последовательностью выполнения операторов. Для этого служат т. н. управляющие операторы.

К управляющим операторам относятся операторы условного перехода и операторы безусловного перехода. Операторы условного перехода – это операторы, определяющие действительную последовательность выполнения операторов программы на основе анализа одного или нескольких условий. В зависимости от того, какое значение (**True** или **False**) будет получено при проверке некоторого заданного условия, VBA определит в операторе условного перехода, какую из ветвей программного кода надо выполнять. Оператор безусловного перехода – это оператор, который просто указывает программе следующий оператор, который должен быть выполнен после данного оператора безусловного перехода.

Оператор выбора Select Case используется в том случае, когда необходимо проверить различные условия, в которых используется одно и то же значение.

Цикл – это фундаментальный компонент любого языка программирования (в том числе и VBA). Под циклом понимается такой оператор или последовательность операторов, которые по ходу выполнения программы могут при необходимости выполняться многократно. Для организации циклов язык VBA предоставляет несколько операторов.

Язык VBA является объектно-ориентированным, т. е. поддерживает такое понятие, как объект. Любое программное приложение, поддерживающее среду программирования VBA, должно включать совокупность объектов. Объектом называют программную конструкцию, которая позволяет инкапсулировать данные вместе с кодом, предназначенным для обработки этих данных, т. е. позволяет объединить их в нечто целое.

Инкапсуляция – это сокрытие объектом некоторых данных таким образом, чтобы доступ к ним в программе был возможен только через специальные функции и процедуры, предоставляемые самим этим объектом и называемые часто методами. Кроме этого, к данным объекта возможен доступ посредством специальных процедур – свойств, которые выглядят с точки зрения программы подобно переменным. Некоторые свойства объекта при необходимости можно изменять – такие свойства называются доступными для чтения и записи. Другие же свойства позволяют только считывать свои значения. Такие свойства объекта называются доступными только для чтения. Многие объекты имеют свойства, значения которых принимаются по умолчанию – они присваиваются этим свойствам при создании объекта, когда для них не задается какое-либо конкретное значение. В каждом VBA-приложении присутствует свой уникальный набор объектов с их собственными свойствами и методами. Во всех VBA-приложениях присутствуют объекты, представляющие экранные формы и обеспечивающие отображение информации, а также объекты отдельных компонентов этих форм, предоставляющие средства обработки отображаемой информации – панели инструментов и отдельные кнопки на этих панелях, меню, переключатели, поля ввода и т. д.

Коллекция объектов – это группа связанных объектов, которые при работе могут использоваться как некое единое целое. Каждый объект в такой коллекции называется элементом этой коллекции. Коллекции используются во всех тех случаях, когда в программе необходимо манипулировать более чем одним объектом. Сами коллекции являются объектами. При необходимости сослаться на член коллекции можно, указав его позицию в коллекции (в этом случае указывается индекс, начиная с 1) или имя требуемого объекта.

Одним из важнейших понятий объектно-ориентированного программирования является класс. Это понятие описывает типовую структуру элементов, представляемых объектами. В системе хранится программное описание каждого известного ей класса, на основании которого при необходимости создаются объекты. Класс определяет данные, а также набор свойств и методов его объектов. Каждый конкретный объект данного класса содержит свои данные.

Работа с объектами проводится с использованием ссылок. Объектная ссылка представляет собой переменную, хранящую адрес, определяющий то место в памяти, где размещаются внутренние данные объекта. На один и тот же объект можно создать несколько ссылок, но в памяти всегда будет присутствовать только одна копия данных этого объекта. Прежде чем использовать ссылку на объект в программе, необходимо создать переменную, в которой эта ссылка будет храниться, затем создать сам объект, получив на него ссылку.

# 2.2 Темы рефератов о VBA

1 Создание форм на VBA в среде Excel.

- 2 Объектно-ориентированное программирование на VBA в среде Excel.
- 3 Работа с интерфейсом Excel средствами VBA.
- 4 Средства работы с файлами в языке VBA.
- 5 История развития языка VBA.

# 2.3 Темы рефератов о методах вычислений

1 Численное решение дифференциальных уравнений первого порядка и их систем.

2 Численные методы безусловной оптимизации.

3 Численные методы дифференцирования и интегрирования.

4 Численные методы приближения функций.

### 3 Тема № 4. Математические программные пакеты

#### 3.1 Краткие сведения о Mathcad

Mathcad – программа, которая относится к классу приложений, называемых PSE (Problem Solution Environment – программная среда для решения). Это подразумевает, что ее работа не определяется однозначно действиями пользователя (как, например, в текстовых редакторах), а является в большей степени результатом работы встроенных алгоритмов.

Приложение Mathcad, принадлежащее теперь компании РТС, – самый популярный из компьютерных математических пакетов. С его помощью можно решать самые разные математические задачи и оформлять результаты расчетов на высоком профессиональном уровне.

Интерфейс программы Mathcad включает заголовок окна, строку меню, панели инструментов, рабочий лист (рабочую область) документа, строку состояния и окно трассировки.

К основным элементам математических выражений в Mathcad относятся числа, переменные, константы, операторы и функции.

Имена переменных, констант и функций могут состоять из любого количества допустимых символов: латинских и греческих букв, цифр и специальных символов («`», «\_», «%»). Имя не может начинаться цифрой. Прописные и строчные буквы различаются.

Некоторые имена в Mathcad зарезервированы под системные переменные, которые называются встроенными константами. Встроенные константы делятся на два типа: математические, хранящие значения некоторых общеупотребительных специальных математических символов, и системные, определяющие работу большинства численных алгоритмов, реализованных в Mathcad.

Основные математические операторы доступны на панели инструментов «Калькулятор». Некоторые из них можно набирать с помощью клавиатуры, например: <+>, <-> и т. п.

Для присваивания числа переменной служит оператор определения «:=», который отображается после набора клавиш <Shift>+<:>.

Значением переменной может быть не только число, но и строка, состоящая из любой последовательности символов, заключенной в кавычки. Кавычки вводятся после оператора «:=» клавишами <Shift>+<">.

Переменной также может быть присвоен диапазон значений. Такая переменная называется ранжированной. Диапазон указывается специальным символом «..». Этот символ набирается клавишей <;>.

Переменную вообще можно определять любым допустимым математическим выражением, в т. ч. с использованием векторов, матриц, тензоров, других переменных. При этом для определения переменой при помощи другой переменной, последняя должна быть определена в рабочей области выше или левее. Это относится к оператору «Определение», т. е. «:=», который называется локальным. Существует также оператор глобального определения

переменной – «≡». Он доступен на панели инструментов «Вычисление», а также может быть набран клавишами <Shift>+<~>. Этот оператор служит для определения переменной, которая может использоваться для определения другой переменной, расположенной в любом месте рабочей области.

Вычисление выражения осуществляется оператором «=», который называется «Рассчитать численно».

Точностью вычислений при помощи оператора «=» можно управлять командой, доступной в меню: «Формат»\«Результат».

Вычисление выражения с непосредственно задаваемой точностью осуществляется оператором «•—», который называется «Символьный расчет с ключевым словом» и доступен на панелях инструментов «Вычисление» и «Символьные», а также может быть набран клавишами <Ctrl>+<>>. Ключевым словом здесь является «float,n», где n – это число знаков после запятой.

В Mathcad доступно много стандартных функций. Некоторые из них можно набирать при помощи клавиш, например: sin, cos и т. п. Ряд функций доступно на панели инструментов «Калькулятор». Все имеющиеся функции доступны в меню: «Вставка»\«Функция». Эта команда активирует диалог «Вставка функции».

Каждая формула ограничена в рабочей области т. н. математической областью, в пределах которой доступны инструменты редактора формул.

Основные элементы интерфейса редактора формул Mathcad:

– указатель мыши – играет обычную для приложений Windows роль, следуя за движениями мыши;

– курсор – обязательно находится внутри документа в одном из трех видов:

a) курсор ввода – крестик красного цвета, который отмечает пустое место в документе, куда можно вводить текст или формулу;

б) линии ввода – горизонтальная и вертикальная линии синего цвета, выделяющие в тексте или формуле определенную часть;

в) линия ввода текста – красная вертикальная линия, аналог линий ввода для текстовых областей;

– местозаполнители – появляются внутри незавершенных формул в местах, которые должны быть заполнены:

а) местозаполнитель символа – черный прямоугольник;

б) местозаполнитель оператора – черная прямоугольная рамка.

Перемещать линии ввода в пределах формулы можно одним из двух способов:

1) щелкая в нужном месте мышью;

2) нажимая на клавиатуре клавиши: стрелки, пробел и <Ins>:

– клавиши со стрелками имеют естественное назначение, переводя линии ввода вверх, вниз, влево или вправо;

– клавиша <Ins> переводит вертикальную линию ввода с одного конца горизонтальной линии ввода на противоположный;

– пробел предназначен для выделения различных частей формулы.

Удаление элементов выражения осуществляется клавишами <Del> и <Backspace>.

Если щелкнуть левой клавишей мышки по математической области, то становится активным математический редактор. Такую математическую область можно перемещать. Для этого следует навести курсор на рамку, нажать левую клавишу мышки и, удерживая ее нажатой, перемещать мышку.

Для удаления математической области ее необходимо выделить прямоугольником и нажать клавишу <Del>.

Кроме математических областей, в Mathcad можно создавать текстовые области. Для этого можно воспользоваться клавишами <Shift>+<">.

Для представления матриц в пакете Mathcad используются массивы двух типов: одномерные и двумерные. Одномерный массив с одним столбцом или одной строкой называется вектором. Двумерный массив называется собственно матрицей. Стандартные функции и операторы для работы с векторами используют именно вектор-столбцы.

Массив может быть присвоен переменной, которая может участвовать в дальнейшем в матричных выражениях.

Номер элемента матрицы называется индексом. Индексы могут иметь только целочисленные значения. Они могут начинаться с нуля или единицы в зависимости от системной переменной ORIGIN, которая может быть равна 0 или 1 соответственно. В случае двумерных матриц индекс состоит из номера строки и номера столбца, разделенных запятой. В случае векторов индекс означает или номер строки (для вектор-столбца), или номер столбца (для вектор-строки). Направление нумерации: слева направо и сверху вниз.

Для задания индексов на панели «Матрица» предусмотрена специальная кнопка «Индекс». Перейти к записи индекса можно также с помощью клавиши <[>.

С матрицами могут производиться различные операции. Основные операторы представлены на панели инструментов «Матрица».

В Mathcad можно как складывать матрицы, так и вычитать их друг из друга. Для этих операторов применяются стандартные символы «+» и «-» соответственно. Матрицы должны иметь одинаковую размерность. Каждый элемент суммы двух матриц равен сумме соответствующих элементов матрицслагаемых. Для того чтобы изменить знаки элементов матрицы, достаточно ввести перед ней знак минуса, как перед обычным числом.

Кроме сложения матриц, Mathcad поддерживает операцию сложения матрицы со скаляром. Каждый элемент результирующей матрицы равен сумме соответствующего элемента исходной матрицы и скалярной величины.

При умножении следует помнить, что матрицу размерности MxN допустимо умножать только на матрицу размерности NxP. В результате получается матрица размерности MxP. Чтобы ввести символ умножения, нужно нажать клавишу со звездочкой <\*>.

Аналогично сложению матриц со скаляром определяется умножение и деление матрицы на скалярную величину. Символ умножения вводится так же, как и в случае умножения двух матриц. На скаляр можно умножать матрицу любой размерности.

Модуль вектора по определению равен квадратному корню из суммы

квадратов его элементов. Оператор берется на панели «Калькулятор».

Скалярное произведение векторов определяется как скаляр, равный сумме попарных произведений соответствующих элементов. Векторы должны иметь одинаковую размерность.

Для вычисления определителя матрицы следует пользоваться оператором на панели «Матрица», который отображается так же, как и модуль.

Рангом матрицы называют наибольшее натуральное число k, для которого существует не равный нулю определитель k-го порядка подматрицы, составленной из любого пересечения k столбцов и k строк матрицы. Для вычисления ранга в Mathcad предназначена функция rank.

Поиск обратной матрицы возможен, если матрица квадратная и ее определитель не равен нулю. Произведение исходной матрицы на обратную по определению является единичной матрицей. Для ввода оператора поиска обратной матрицы следует воспользоваться оператором «Обращение» на панели инструментов «Матрица».

Для выделения из матрицы столбца применяется оператор «Столбец матрицы» на панели «Матрица».

Для выделения подматрицы используется встроенная функция submatrix(A,ir,jr,ic,jc), возвращающую часть матрицы A, находящуюся между строками ir,jr и столбцами ic,jc включительно.

Для того чтобы составить из двух или более матриц одну, в Mathcad предусмотрена пара матричных функций:

– augment(A,B,C,...) – матрица, сформированная слиянием матрицаргументов слева направо;

– stack(A,B,C,...) – матрица, сформированная слиянием матриц-аргументов сверху вниз, где A,B,C,... – векторы или матрицы соответствующего размера.

Подобно присваиванию числовых значений переменным, можно определить функции пользователя одного или нескольких аргументов.

Для построения графика функции используются команды панели «График».

На графике имеются местозаполнители для ввода аргумента, функции и пределов изменения аргумента и функции. Если используются несколько функций, то они набираются через запятую. Если эти функции зависят от разных аргументов, то последние набираются в соответствующем местозаполнителе также через запятую.

Форматирование графиков осуществляется при помощи специального диалогового окна, которое активируется из меню: «Формат»——«График».

В местозаполнитель трехмерного графика следует ввести либо имя функции двух переменных, либо имя матричной переменной, которая задаст распределение данных на плоскости ХҮ.

Дифференцирование и интегрирование осуществляется при помощи инструментов, расположенных на панели «Математический анализ».

Оператор «Производная» может использоваться для аналитического определения производной.

Для решения нелинейных уравнений и их систем в Mathcad применяется специальный вычислительный блок Given/Find: после ключевого слова Given

записываются уравнения с использованием логического знака равенства (панель «Булева алгебра» или сочетание клавиш <Ctrl>+<=>), а затем используется встроенная функция Find(x1,...,xM) для решения системы уравнений относительно переменных x1,...,xM.

Для аналитического решения нелинейного алгебраического уравнения предусмотрена и иная запись, связанная с использованием ключевого слова solve.

### 3.2 Темы рефератов о математических программных пакетах

1 Программирование в среде Mathcad.

2 Символьные преобразования в Mathcad.

3 Средства оптимизации в Mathcad.

4 Решение дифференциальных уравнений в среде Mathcad.

5 История развития математических программных пакетов.

6 Возможности математического программного пакета Maple.

7 Возможности математического программного пакета MATLAB.

# Список литературы

1 Козлов, А. Ю. Статистический анализ данных в MS Excel : учебное пособие / А. Ю. Козлов, В. С. Мхитарян, В. Ф. Шишов. – Москва : ИНФРА-М, 2019. – 320 с.

2 Гуриков, С. Р. Введение в программирование на языке Visual Basic for Applications (VBA) : учебное пособие / С. Р. Гуриков. – Москва : ИНФРА-М, 2020. – 317 с.

3 Карманов, Ф. И. Статистические методы обработки экспериментальных данных с использованием пакета MathCad : учебное пособие / Ф. И. Карманов, В. А. Острейковский. – Москва : КУРС ; ИНФРА-М, 2018. – 208 с.