

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ГРАЖДАНСКОЙ АВИАЦИИ» (МГТУ ГА)**

**Кафедра прикладной математики
В.М. Коновалов**

**ИНФОРМАТИКА
И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ**

**ПОСОБИЕ
по изучению дисциплины и выполнению
контрольных и лабораторных работ**

*для студентов I курса
направления 25.03.02
заочной формы обучения*

Москва-2015

ББК 6Ф6.5

К64

Рецензент канд. техн. наук, доц. А.Г. Роцин

Коновалов В.М.

К64 Информатика и информационные технологии: пособие по изучению дисциплины и выполнению контрольных и лабораторных работ. - М.: МГТУ ГА, 2015. - 40 с.

Данное пособие издается в соответствии с рабочей программой учебной дисциплины «Информатика и информационные технологии» по Учебному плану для студентов I курса направления 25.03.02 заочной формы обучения.

Рассмотрено и одобрено на заседаниях кафедры 20.01.2015 г. и методического совета 20.01.2015 г.

Подписано в печать 02.03.2015 г.

Печать офсетная
2,33 усл.печ.л.

Формат 60x84/16
Заказ № 1965/

2,17 уч.-изд. л.
Тираж 100 экз.

Московский государственный технический университет ГА
125993 Москва, Кронштадтский бульвар, д.20
Редакционно-издательский отдел
125493 Москва, ул. Пулковская, д.6а

© Московский государственный
технический университет ГА, 2015

Содержание

Предисловие.....	4
Раздел 1. Методические указания к изучению дисциплины	5
1.1. Общая характеристика процесса изучения дисциплины	5
1.2. Распределение часов по видам занятий	5
1.2.1. Содержание лекций.....	5
1.2.2. Содержание практических занятий	7
1.2.3. Содержание лабораторных работ	7
1.2.4. Содержание контрольных работ.....	8
1.3. Промежуточная аттестация.....	9
1.4. Учебно-методическое и информационное обеспечение дисциплины	11
1.5. Материально-техническое обеспечение дисциплины.....	11
Раздел 2. Методические указания к выполнению контрольных и лабораторных работ	12
2.1. Основные определения.....	12
2.2. Математические функции	13
2.3. Типы данных.....	14
2.4. Варианты заданий	16
2.4.1. Задание № 6. Линейные алгоритмы	16
2.4.2. Задание № 7. Ветвящиеся алгоритмы	20
2.4.3. Задание № 8. Циклические алгоритмы	25
2.4.3.1. Задача № 8.1	25
2.4.3.2. Задача № 8.2.....	29
2.4.4. Задание № 9. Алгоритмы смешанного типа.....	32
2.5. Интерактивные приложения	36
2.5.1. Процедуры	36
2.5.2. Интерфейсы	38
2.6. Вопросы для самопроверки при подготовке к аттестации	40

Предисловие

В пособии приведена общая характеристика процесса изучения дисциплины "Информатика и информационные технологии", а так же рассмотрены конкретные вопросы, возникающие у студентов-заочников при выполнении контрольной и лабораторных работ. Необходимость в написании данного пособия возникла в результате изменений в Учебном плане данной дисциплины, обусловивших переход к 2х-семестровому курсу обучения.

Изложение материала пособия относится к двум разделам рабочей программы дисциплины: "Арифметические и логические основы информатики" и "Алгоритмизация и программирование", как наиболее трудоемким разделам. При этом в пособии предметно рассматриваются вопросы второго из названных разделов, учитывая, что методические указания к изучению первого из них подробно изложены в [5]. Предполагается, так же, что остальные разделы рабочей программы, изучаются студентом в процессе семестровой работы.

В первом разделе данного пособия приведены: распределение общего объема часов на изучение дисциплины по видам занятий, из которого следует, что основная учебная нагрузка студента – это самостоятельная работа (СРС) в течение семестра; тематика аудиторных лекционных занятий; содержание и последовательность выполнения работы на практических, лабораторных занятиях во время сессии; содержание контрольной работы.

Особое внимание обращено на те моменты работы, которые связаны с оформлением ее результатов и их представлением для прохождения аттестации (зачет, экзамен).

Во втором разделе пособия приведены необходимые сведения справочного характера, применительно к рассматриваемым, далее, примерам программирования вариантов заданий контрольной работы, выполняемым в среде VBA (Visual Basic for Applications), используемой для программирования в приложениях Microsoft Office. Этому предшествует этап выполнения студентами, в первом семестре, индивидуальных заданий по алгоритмизации вычислительных задач, с отчетностью в среде QBasic [5].

На данном этапе обучения требуется подтверждение умений не столько собственно программирования, как процесса написания кода на некотором языке программирования, сколько демонстрации умений правильной алгоритмизации задачи.

Изложение учебно-методического материала завершается рассмотрением информационной технологии, связанной с разработкой интерактивных интерфейсов пользовательских приложений, предназначенных как для автоматизации работы с документами, электронными таблицами, диаграммами, презентациями и т.д., так и для разработки довольно сложных приложений обработки данных с использованием диалоговых окон, обеспечивающих пользователей современными интерфейсными средствами для работы с Windows-приложениями.

Раздел 1. Методические указания к изучению дисциплины

1.1. Общая характеристика процесса изучения дисциплины

Изложение учебного материала выполняется в соответствии с программой, которая является приложением к рабочей программе дневного отделения для учета специфики обучения на заочном отделении.

ПРИЛОЖЕНИЕ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

"ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ"

направление подготовки **25.03.02** - *Техническая эксплуатация авиационных электросистем и пилотажно-навигационных комплексов*, квалификация "бакалавр"

шифр специальности (направления подготовки)

Заочная форма обучения

Курс обучения	I
Общий объём часов	360 час.
Объём учебных часов	38 час.
Самостоятельная работа студента	322 час.

1-я сессия:

<i>Лекции</i>	<i>10 час.</i>
<i>Практические занятия</i>	<i>6 час.</i>
<i>Лабораторные занятия</i>	<i>8 час.</i>
<i>Зачёт</i>	

2-я сессия:

<i>Лекции</i>	<i>4 час.</i>
<i>Практические занятия</i>	<i>2 час.</i>
<i>Лабораторные занятия</i>	<i>8 час.</i>
<i>Контрольная работа</i>	<i>1</i>
<i>Экзамен</i>	

1.2. Распределение часов по видам занятий

1.2.1. Содержание лекций

Тема 1. Введение в информатику. [8, Гл. 1-4, 10-12, 17] (2 час)

Предмет информатики, задачи, цели курса. Содержание дисциплины. Методические рекомендации по выполнению самостоятельной работы. Содержание контрольной работы, методика ее выполнения. Выполнение лабораторного практикума. Рекомендуемые источники информации.

1-я сессия:

Тема 2. Автоматизация работы с документами на базе офисного программирования в Visual Basic для приложений (VBA). [1, Гл. 1-3; 8, Гл. 12, 20; 9, Лек.2-3] (2 час.)

Интерфейс инструментальной интегрированной среды программирования – IDE (Integrated Development Environment). Элементы синтаксиса языка Visual Basic. Событийное управление выполнением программы в объектно-ориентированной среде. Возможности VBA. Элементы проекта VBA. Выбор основного приложения VBA. Расширение функциональных возможностей основного приложения. Работа в редакторе VBA.

Тема 3. Основные сведения о технологии создания приложений на VBA. [1, Гл. 4-6, 13; 9, Лек.4-5] (2 час)

Создание проекта VBA. Открытие существующего проекта. Работа с окном проекта. Использование окна свойств. Добавление модуля в проект. Использование окна модуля. Создание форм. Добавление форм в проект. Импорт и экспорт файлов. Переключение в основное приложение. Запуск форм, процедур и макросов. Сохранение изменений. Задание параметров редактора VBA. Установка свойств проекта. Элементы управления в приложениях VBA. Сохранение кода в модулях и проектах. Соглашения по именованию.

Тема 4. Создание процедур VBA. [1, Гл. 7-9; 9, Лек.8-10] (2 час)

Процедуры Sub, Function; автоматическая запись процедуры; создание процедуры в редакторе VBA; структура процедуры. Рекомендации по вводу кода. Инструкции, функции и выражения. Рекомендации по стилю программирования. Объектно-ориентированное программирование в VBA. Основные сведения об объектах. Объекты в VBA. Использование свойств и методов объектов.

Тема 5. Разработка интерфейсов пользовательских приложений, отладка программ и обработка ошибок. [1, Гл. 10, 12; 9, Лек. 11-14] (2 час)

Синтаксис объектов VBA. Программирование объектов приложения. Событийное управление работой приложения с помощью объектов управления на интерактивной форме. Элементы отладки программ в VBA и обработка ошибок выполнения.

2-я сессия:

Тема 6. Компьютерные сети. [2, Гл. 2-3; 8, Гл. 8] (2 час)

Задачи надежного обмена двоичными сигналами при передаче данных по линиям связи. Проблемы объединения нескольких компьютеров в сеть: топология физических связей; организация совместного использования линий связи; адресация компьютеров. Базовые технологии локальных сетей. Ethernet – пример стандартного решения сетевых проблем.

Тема 7. Информационные технологии межсетевое взаимодействия. [2, Гл. 4; 8, Гл. 9, 19] (2 час)

Многоуровневый подход в организации сетевого взаимодействия компьютеров: протокол, интерфейс, стек протоколов. Модель ISO взаимодействия открытых систем -OSI. Функции уровней эталонной модели. Проблемы стандартизации. Стандартные стеки коммуникационных протоколов.

1.2.2. Содержание практических занятий

Практические занятия, во время 1-й экзаменационной сессии, направлены на закрепление **знаний** и проверку **умений** разрабатывать (в среде Quick Basic – QB-45) алгоритмы и программировать (в среде Visual Basic for Applications – VBA) вычислительные процессы. Приобретенные знания и умения являются результатом работы студента в течение первого семестра.

Во время 2-й сессии – на закрепление и проверку **умений** использования технологии разработки интерактивных приложений, автоматизирующих работу с деловой информацией.

1-я сессия: [3; 4; 5; 7]

ПР 1. (2 час.) QB-45: Базовые алгоритмические структуры.

ПР 2. (2 час.) VBE: Изучение приемов работы в инструментальной среде.

ПР 3. (2 час.) VBA: Создание процедур.

2-я сессия: [3; 4; 5]

ПР 4. (2 час.) Технология разработки приложений в инструментальной среде VBE

1.2.3. Содержание лабораторных работ

В период экзаменационных сессий предусмотрено выполнение студентами-заочниками четырех лабораторных работ. Объем каждой работы – 4 часа. Общий объем – 16 часов.

Целью выполнения работ является выработка **навыков** использования в разрабатываемых программах базовых алгоритмических конструкций, достаточных для реализации любого вычислительного алгоритма. Программирование рассматривается как компьютерный эксперимент по реализации алгоритмических структур, без которого развитие алгоритмического мышления является практически невозможным.

1-я сессия: [3; 4; 5; 7]

ЛР 1. (4 час.) Алгоритмические структуры в программах на QB-45.

ЛР 2. (4 час.) Алгоритмические структуры в программах на VBA.

2-я сессия: [3; 4; 5]

ЛР 3. (4 час.) Разработка интерактивных пользовательских приложений: процедуры.

ЛР 4. (4 час.) Разработка интерактивных пользовательских приложений: интерфейсы

1.2.4. Содержание контрольных работ

В соответствии с Учебным планом, по дисциплине предусмотрено выполнение во 2-м семестре одной **контрольной работы**, заключающейся:

1) в письменном решении задач по материалу раздела "Арифметические и логические основы информатики" [5; 6, Лк. 6-9].

2) в разработке алгоритмов и написании программ на языке программирования Visual Basic v.6.0 для их реализации в инструментальной среде программирования VBE – по материалу раздела "Алгоритмизация и программирование" [3; 5; 9]. Контрольная работа выполняется в процессе самостоятельной работы студента (СРС) **во втором семестре** и должна быть подготовлена **к началу 2-й сессии**.

Примечание 1. Контрольная работа представляется студентом в Деканат в письменной (на листах офисной бумаги для печати формата А4) форме, и подлежит регистрации методистом за две недели до начала 2-й сессии.

Контрольная работа содержит 9 самостоятельно выполняемых заданий. В данном пособии сохранены нумерация и содержание заданий в том виде, в каком они приведены в [5].

I. Арифметические и логические основы информатики

Задание 1. Перевод чисел из десятичной системы счисления в систему счисления с основанием "b".

Задание 2. Перевод чисел из системы счисления с основанием "b" в десятичную систему счисления.

Задание 3. Перевод чисел из СС с основанием 8 в СС с основанием 16 (или обратно).

Задание 4. Сложение целых двоичных чисел в обратном или дополнительном коде.

Задание 5. Сложение чисел с плавающей запятой.

II. Алгоритмизация и программирование

Задание 6. Программирование простых выражений.

Задание 7. Программирование ветвящихся алгоритмов.

Задание 8. Программирование циклических алгоритмов.

Задание 9. Программирование алгоритмов смешанного типа.

1.3. Промежуточная аттестация

1-я сессия:

Для получения ЗАЧЁТА по дисциплине "Информатика и информационные технологии" необходимо во время 1-й сессии:

1. По результатам самостоятельной работы в **1-м семестре** представить преподавателю **Отчет** в электронной форме с выполненными, в соответствии с вариантом, **заданиями №№ 6÷9** из методического пособия [5]: **Андреева Т.И. Информатика. Пособие по изучению дисциплины и выполнению контрольных работ для студентов заочной формы обучения М.: МГТУ ГА, 2008.** Там же приведены правила определения номера **варианта** задания:

Для демонстрации результатов изучения **принципов алгоритмизации** вычислительных задач, использовать систему программирования **QBASIC v.4.5**, как наиболее приемлемую **на данном** этапе обучения.

Примечание 2. **Отчет** представляется студентом **преподавателю** на практическом/лабораторном занятии (во время **1-й сессии**) в **электронной** форме в виде текстового **именованного** файла, **сохраненного на флэш-памяти**, подготовленного в текстовом редакторе MS Word (шрифт – Times New Roman, размер – 14, межстрочный интервал – одинарный). Например, если отчет выполнен студентом Ивановым И.И., имеющим номер зачётной книжки АК-111556, то именованный файл будет иметь следующий вид: **Отчет-Иванов ИИ АК-111556 .doc** (или ***.docx**).

Структура отчета:

- титульный лист (по стандарту) с адресными данными студента;
- описание варианта задания;
- схема (в стандартных графических изображениях блоков) алгоритма решения задачи;
- исходный текст (напечатанный вручную или в виде экранной копии окна с **исходным** модулем) программы на языке **QBASIC**;
- **экранный** копия соответствующего заданию окна с **результатом** вычислений;
- использованные источники информации.

Демонстрационные (исполняемые) файлы подготавливаются в системе программирования QB-45 и именуется **в соответствии** с номером и вариантом задания. Например, выполненное задание №7 по варианту 26 следует сохранить на флэш-памяти в файле **lab726.bas**, а задание № 8.2а по этому же варианту – в файле, имя которого **lab82a26.bas**.

2. Подтвердить работоспособность разработанных алгоритмов для заданий №№ 6-9 в среде программирования QB-45 на лабораторных/практических занятиях 1-й сессии.

В оставшееся (после выполнения п.2) время, ознакомиться с основами работы в инструментальной среде программирования VBE. Используйте, в качестве host-приложения, компоненту **Excel** из пакета MS Office [1, Гл. 13; 8, Гл. 12] на лабораторных и практических занятиях.

2-я сессия:

Для сдачи ЭКЗАМЕНА по дисциплине "Информатика и информационные технологии" необходимо во время 2-й сессии:

1. По результатам самостоятельной работы во **2-м семестре**, представить методисту деканата (для регистрации) выполненную контрольную работу по вышеуказанным заданиям №№ 1÷9 (в соответствии с вариантом)

Для демонстрации (на аудиторных занятиях, преподавателю) результатов изучения **принципов VBA-программирования** вычислительных задач (задания №№ 6÷9) использовать инструментальную среду программирования VBE офисного пакета MS Office.

2. Выполнение контрольной работы, заключается:

1) в письменном решении задач по материалу раздела "Арифметические и логические основы информатики" (задания №№ 1÷5 из методического пособия [5]).

2) в написании программ, по разработанным ранее (в среде QB-45) алгоритмам, на языке программирования VBA для их реализации в инструментальной среде программирования VBE – по материалу раздела "Алгоритмизация и программирование" (задания №№ 6÷9).

Примечание 3. При выполнении каждого задания следует придерживаться следующих требований:

- при решении задач (**задания №№ 1÷5**) следует записать полный текст задания для своего варианта. Подробно изложить процесс решения с промежуточными результатами так, как это описано в сопровождающих примерах.

- при решении задач (**задания №№ 6÷9**) требования к оформлению результатов работы с программным продуктом аналогичны тем, которые сформулированы в **Примечании 2** для оформления Отчета (по 1-ому семестру) с тем отличием, что все сказанное относится к работе на языке программирования VBA. в среде VBE.

Подготовленный и отредактированный, с учетом требований к оформлению, текстовый файл с контрольной работой сохраняется на флэш-памяти (например, как файл с именем **KP1-Иванов III АК-111556.docx**), распечатывается на бумаге и передается в деканат для регистрации.

Примечание 4. Демонстрационные (исполняемые) файлы, подготовленные в инструментальной среде VBE, следует именовать **в соответствии** с номером задания и вариантом. **Например**, выполненное задание №7 по варианту 26 следует сохранить на флэш-памяти в файле **lab726.xls** (или ***.xlsx, *.xlsm**), а задание № 8.2а – в файле, имя которого **lab82a26.xls** (или ***.xlsx, *.xlsm**).

3. Подтвердить работоспособность разработанного приложения для одного из заданий №№ 6÷9 (по выбору) в среде программирования VBE, используя в качестве host-приложения офисное приложение **Excel** пакета MS Office на лабораторных и практических занятиях 2-й сессии.

1.4. Учебно-методическое и информационное обеспечение дисциплины

№	Автор	Наименование, издательство, год издания
1	2	3
Основная литература		
1	Слепцова Л.Д.	Программирование на VBA в Microsoft Office 2010. – М.: Вильямс, 2010. – 432 с.
2	Олифер В.Г., Олифер Н.А.	Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 4-е изд. – СПб.: ПИТЕР, 2010. – 944 с.
Уч.-метод. литература для выполнения практических работ		
3	Кузьменко В.Г.	VBA. Эффективное использование. Издательство: Бином-Пресс, 2012. – 617 с.
4	Культин Н. Б.	Visual Basic. Освой самостоятельно. Издательство: БХВ-Петербург, 2009. – 488 с.
Уч.-метод. литература для выполнения контрольных работ		
5	Андреева Т.И.	Информатика. Пособие по изучению дисциплины и выполнению контрольных работ для студентов заочной формы обучения М.: МГТУ ГА, 2008.
6	Чуканов В.О., Гуров В.В.	Логические и арифметические основы и принципы работы ЭВМ. - Курс лекций. <i>Дистанционное обучение</i> . - [Электронный ресурс], URL: http://www.intuit.ru/department/hardware/archsys/ (Дата обращения 15.01.2015)
Уч.-метод. литература для выполнения лабораторных работ		
7	Андреева Т.И., Кишенский С.Ж.	Информатика. Часть II. Пособие по выполнению лабораторных работ для студентов I курса дневного обучения М.: МГТУ ГА, 2009.
Дополнительная литература		
8	Симонович С.В. и др.	Информатика. Базовый курс. Учебник для вузов. 2-е изд. - СПб: Питер, 2004 – 640 с.
9	Биллиг В.А.	Основы офисного программирования и язык VBA. – Курс лекций. <i>Дистанционное обучение</i> . – [Электронный ресурс], URL: http://www.intuit.ru/department/office/vba2000/ (Дата обращения 15.01.2015)

1.5. Материально-техническое обеспечение дисциплины

Учебный класс, оборудованный персональными компьютерами, на которых установлены:

- система программирования Quick Basic v. 4.5;
- пакет прикладных программ MS Office.

Компьютеры учебного класса являются рабочими станциями сегмента локальной сети Университета, предоставляющей доступ к электронным ресурсам университетской библиотеки, а также широкополосный доступ в Интернет для реализации режима дистанционного обучения.

Раздел 2. Методические указания к выполнению контрольных и лабораторных работ

2.1. Основные определения

Решение задачи с помощью компьютера состоит из следующих этапов [5]:

- математической формулировки условия задачи;
- выбора численного метода ее решения;
- разработка алгоритма;
- составление программы на алгоритмическом языке;
- отладка программы.

В предлагаемых ниже заданиях условия задач уже представлены в математической формулировке с указанием численного метода решения и необходимость в выполнении первых двух этапов отпадает. При выполнении остальных из вышеперечисленных этапов, используется следующая терминология.

Алгоритм – это последовательность действий, однозначно определяющих процесс преобразования исходных и промежуточных данных в результат решения задачи. Форма представления алгоритма может быть как текстовой, так и графической – в виде схемы.

Решение всего многообразия задач может быть сведено к трем типам алгоритмов: *линейному, разветвляющемуся и циклическому*. Чаще встречается комбинация этих типов.

Линейный алгоритм – алгоритм, в котором к результату решения задачи приводит *последовательное* выполнение действий.

Алгоритм разветвляющейся структуры – алгоритм, в котором последовательность выполнения действий зависит от каких-либо *условий*.

Циклический алгоритм – алгоритм, в котором для получения результата требуется многократное (циклическое) *повторение* ряда действий.

Программа – это последовательность строк, описывающих на языке программирования алгоритм решения поставленной задачи.

Строка может содержать один или несколько операторов, а также комментарии (пояснения).

Оператор представляет собой строго формализованное указание на выполнение конкретного действия.

Visual Basic (VB). Визуальная версия языка программирования высокого уровня Basic. Язык Visual Basic был разработан специалистами корпорации Майкрософт для создания приложений на базе Microsoft Windows.

Visual Basic for Applications (VBA) – диалект языка VB, применяемый для расширения функциональных возможностей офисных приложений, входящих в пакет прикладных программ MS Office.

Проект/Приложение VBA – совокупность модулей.

Модуль. Совокупность описаний, инструкций и процедур, сохраненная под общим именем. Существуют модули двух типов стандартный модуль и модуль класса.

Процедура – поименованный набор описаний и инструкций.

Описание – это инструкция, которая используется для представления переменных, констант, типов данных, ссылок на внешние функции, хранящиеся в библиотеках динамической компоновки.

Инструкция – единица программного кода, которая является операцией или описанием.

Макрос. Набор команд и инструкций, сгруппированных в одну команду для автоматического выполнения какой-либо задачи. Запись макросов автоматически производится макрорекодером на языке программирования Visual Basic для приложений – VBA.

Редактор Microsoft Visual Basic (VBE-Visual Basic Editor). Инструментальная среда программирования, позволяющая редактировать записанные макросы, а также создавать новые макросы и пользовательские приложения на языке VBA.

Host-приложение – это приложение пакета MS Office, такое как Word, Excel, PowerPoint, Outlook и др., в котором разрабатываются процедуры VBA.

2.2. Математические функции

VBA предоставляет стандартный набор математических функций, приведенный в табл. 2.2.1. В ней N означает любое (из области определения) численное выражение; все аргументы функций являются обязательными, если только не указано обратное.

Таблица 2.2.1

Функции (аргументы)	Возвращает/Действие
Abs(N)	Возвращает абсолютное значение N, что соответствует математической записи – N .
Atn(N)	Возвращает арктангенс числа N, как угол в радианах.
Cint(N)	Возвращает значение, являющееся округлением числа N
Cos(N)	Возвращает косинус угла N, где N – угол, измеренный в радианах.
Exp(N)	Возвращает константу e , возведенную в степень N, что соответствует математической записи e^N (e – основание натуральных логарифмов и

	оно, приблизительно, равно 2,718282).
Fix(N)	Возвращает целую часть N. Fix не округляет число, а отбрасывает любую дробную часть. Если N является отрицательным, Fix возвращает ближайшее отрицательное целое, большее, чем N.
Int(N)	Возвращает целую часть N. Int не округляет число, а отбрасывает любую дробную часть. Если N является отрицательным, Int возвращает ближайшее отрицательное целое, меньшее, чем N.
Log(N)	Возвращает натуральный логарифм N, что соответствует математической записи Ln(N).
Rnd(N)	Возвращает случайное число обычной точности (Single) в интервале $0 \div 1$; аргумент N является необязательным. Используйте функцию Rnd только после инициализации VBA-генератора случайных чисел оператором Randomize в начале процедуры.
Sgn(N)	Возвращает знак числа: -1, если N – отрицательное; 1, если N – положительное; 0, если N равно нулю.
Sin(N)	Возвращает синус угла; N – это угол, измеренный в радианах.
Sqr(N)	Возвращает корень квадратный из N. VBA отображает ошибку времени выполнения, если N – отрицательное.
Tan(N)	Возвращает тангенс угла; N – угол в радианах.

Дополнительные тригонометрические функции можно выводить из базовых математических функций VBA. Например, если необходимо вычислить котангенс угла, то для его нахождения можно использовать формулу $1/\text{Tan}(N)$.

VBA вычисляет натуральные логарифмы. Поэтому, для вычисления логарифма числа N по некоторому основанию b, следует воспользоваться известной из курса математики формулой перехода к натуральным логарифмам: $\text{Log}_b N = \text{Ln } N / \text{Ln } b$. Например, в синтаксисе VBA: $\text{Log}_3(N) = \text{Log}(N) / \text{Log}(3)$.

При вычислениях в задаче 8.1б необходимо, чтобы случайные числа – значения переменной X, принадлежали интервалу [a,b]. Для выполнения этих ограничений, запишите инструкцию присваивания значений, генерируемых функцией Rnd(), переменной X(I) в формате: $X(I) = a + K * \text{Rnd}()$, где a – нижняя граница интервала, а значение K выбирается таким, чтобы при $\text{Rnd}() = 1$ значение выражения в правой части инструкции присваивания было бы равно b – верхней границе интервала [a,b]. В этом случае числам из диапазона [0, 1], генерируемым датчиком случайных чисел, будут соответствовать случайные значения переменной X(I) в интервале [a, b].

2.3. Типы данных

Типы данных – это определенные виды данных, которые VBA сохраняет и может манипулировать ими. Как и подавляющее большинство систем программирования VBA разделяет обрабатываемые данные на числа, текст,

даты и другие типы. В таблице 2.3.1 приведены основные типы, используемые в VBA.

Таблица 2.3.1

Тип данных	Размер в байтах	Описание и диапазон значения
Byte	1	Хранение положительных чисел от 0 до 255
Boolean	2	Хранение логических значений: True, False
Currency	8	Хранение чисел для точных вычислений в диапазоне от -922337203685477,5808 до 922337203685477,5807
Date	8	Хранение даты и времени. Даты от 1.01.100 до 31.12.9999 Время от 00:00:00 до 23:59:59
Double	8	Хранение вещественных чисел двойной точности от $-1,79769313486232 \cdot 10^{308}$ до $-4,94065645841247 \cdot 10^{-324}$ от $4,94065645841247 \cdot 10^{-324}$ до $1,79769313486232 \cdot 10^{308}$
Integer	2	Хранение целых чисел от -32768 до 32767
Long	4	Хранение целых чисел от -2147483648 до 2147483647
Object	4	Используется для доступа к любому объекту, распознаваемому VBA. Сохраняет адрес объекта в памяти, являясь, таким образом, указателем на объект.
Single	4	Хранение вещественных чисел одинарной точности от $-3,402823 \cdot 10^{38}$ до $-1,401298 \cdot 10^{-45}$ от $1,401298 \cdot 10^{-45}$ до $3,402823 \cdot 10^{38}$
String (переменной длины)	10+n	Хранение текста длиной до 2 млрд. символов n – длина строки в символах.
String (фиксированной длины)	n	Хранение текста длиной до 654000 символов n – длина строки в символах.
Variant		Хранение любого типа данных (кроме Object). Диапазон для данных типа Variant зависит от фактически сохраняемых данных. В случае текста диапазон соответствует строковому типу; в случае чисел – какому-либо числовому типу.

2.4. Варианты заданий

2.4.1. Задание № 6. Линейные алгоритмы

Написать VBA-программу вычисления и вывода на экран значения выражения для исходных данных (данные выбрать самостоятельно, в пределах области допустимых значений), вводимых в диалоге.

1. $6y \cdot 2x + 4^y + \log_3(7x^2)$
2. $7,45 e^{xy} + 6,98 \sin(xy)$
3. $a^x [\ln(x)]^y \operatorname{arctg}[(1 - x^3 / y)]$
4. $\operatorname{arctg}(|y|/x) + y^3 - x^2 + 1$
5. $\sin(y) - y^{-3} + |\cos(x)| + x^{0,8}$
6. $x + a / [7bx^2 - 4 / ax^3 + \cos(3 / 4x)] - \sin(x)$
7. $x^3 + 1 / y + \log_3(7x)$
8. $xy / (x+y) \operatorname{arctg}[(1 - x^2 / y^2)^{-1/2}]$
9. $|((x+1)^{1/2} - y^{1/3})| \sin(x)$
10. $6y \cdot 2x^{1/3} + 4xy - \sin(x - 2y) / \cos(b) \cdot (2x - y)$
11. $8^{2/x} - 2^{3x+3} + \operatorname{tg}(x / y)$
12. $\log_3 y + \log_4 x^2 + \log_5(xy)^3 + \operatorname{tg}(x)$
13. $\sin^2(|x + y|) + \cos^2(xy)$
14. $|2x - y| / \ln(x + y) + |\sin(y)|^{-1/5}$
15. $(3x + yx - 6)^{1/2} \cdot \ln(x + y) \cdot \operatorname{tg} y$
16. $(3x + 1)^{3,6} \cdot [y + \cos(x)]^{-2} + \ln(x + y) / \ln(x - y)$
17. $|x| + y [74x(y - 9x)^2]^{-7} \cdot (\operatorname{arctg} x)^{-1/2}$
18. $[y - \sin(x)] / \log_9(xy) + (x - 2y)^{-1/3}$

$$19. 4^{\ln x} + \log_3(7x^2) - \left| \sin(x+y) / \cos(x-y) \right|$$

$$20. 4^x - 10 \cdot 2^{x-1} - \sin(3x / (x-1))$$

$$21. \operatorname{tg}^2(x) + 4y + \log_3(7x^2)$$

$$22. x^{0,8} + \sin(y) - y^{-3} + \left| \cos(x) \right|$$

$$23. y^2 + x^2 + 5xy + 78$$

$$24. \sqrt[3]{\frac{\sin(x+y)}{x(x+1)} + \frac{x - \cos(xy)}{7+y}}$$

$$25. (1/x)^3 + (1/y)^2 + \operatorname{tg}(xy)$$

$$26. [2 \sin^3(\operatorname{arctg} x - \sin(x/2))]]$$

$$27. 2\cos^3 x + 4e^{-x} + [ax + b(zxc)]^4$$

$$28. 3,22y + 6,27 \operatorname{arctg} [(1 - x^2/y^2)^{-1/2}]$$

$$29. 5x + 2 \sin^2(y) - 4 / (x+7,2) + a b c^2$$

$$30. 6,28 x + 3,14 y \cdot \ln\left(\frac{1-x}{1+x}\right)$$

Пример 6. Написать программу вычисления и вывода на экран радиуса описанной окружности R1 и радиуса вписанной окружности R2 правильного многоугольника. Вычислить площадь правильного многоугольника.

Количество сторон многоугольника - *n* и длину его стороны - *a* задать в диалоге.

Для вычисления воспользуемся следующими формулами:

$$R1 = \frac{a}{2 \sin(3,14/n)} \quad - \text{ радиус описанной окружности;}$$

$$R2 = \frac{a}{2 \operatorname{tg}(3,14/n)} \quad - \text{ радиус вписанной окружности;}$$

$$S = \frac{n \cdot a \cdot R2}{2} \quad - \text{ площадь правильного многоугольника.}$$

Алгоритм программы в словесной форме состоит из следующих пунктов:

- начало программы;

- ввод значений переменных **a** и **n**;
- вычисление функции R1, R2 и S;
- вывод значений функций R1, R2 и S;
- окончание программы.

В окне модуля среды VBE запишем исходный текст VBA-программы в виде процедуры:

```
Option Explicit ' Требуется явное объявление переменных модуля
'
' Простые вычисления по формулам. Линейный алгоритм.
'
Sub Radius_And_Area()
    Dim N As Integer
    Dim A As Single
    Dim R1 As Double, R2 As Double, S As Double
    N = Val(InputBox("Количество сторон многоугольника:", "Ввод N"))
    A = Val(InputBox("Длина стороны многоугольника:", "Ввод A"))
    R1 = A / (2 * Sin(3.14 / N))
    R2 = A / (2 * Tan(3.14 / N))
    S = N * A * R2 / 2
    Debug.Print "Результат:"
    Debug.Print "Количество сторон N=" & N
    Debug.Print "Длина стороны A=" & A
    Debug.Print "Радиус описанной окружности R1=" & R1
    Debug.Print "Радиус вписанной окружности R2=" & R2
    Debug.Print "Площадь S=" & S
End Sub
```

В окне **Immediate** среды VBE наблюдаем вывод на печать результатов вычислений:

Результат:

Количество сторон N=6

Длина стороны A=4

Радиус описанной окружности R1=4,0018400248889

Радиус вписанной окружности R2=3,46622613007328

Площадь S=41,5947135608793

Комментарий к вышеприведенному примеру.

Программа на VBA оформлена в виде **процедуры** Sub(), имеющей имя Radius_And_Area. Чтобы было легче обнаруживать ошибки, обусловленные неявным объявлением переменных, VBA предоставляет команду **Option Explicit**. При использовании этой директивы, VBA требует **явного** объявления всех переменных (с помощью оператора **Dim**) перед их использованием. Option Explicit препятствует неявному объявлению переменных где-либо в модуле, содержащем эту команду.

Чтобы установить режим работы компилятора, при котором VBA требует явного объявления для всех переменных в модуле, следует добавить команду Option Explicit **в область объявлений модуля**, то есть в начало модуля перед любыми объявлениями переменных или процедур.

Интерактивный режим **ввода** реализован с помощью функции ввода данных **InputBox()** и функции преобразования данных **Val()**.

Функция **InputBox()** позволяет вводить данные с помощью диалогового окна (рис. 2.4.1) и имеет **полный** синтаксис:

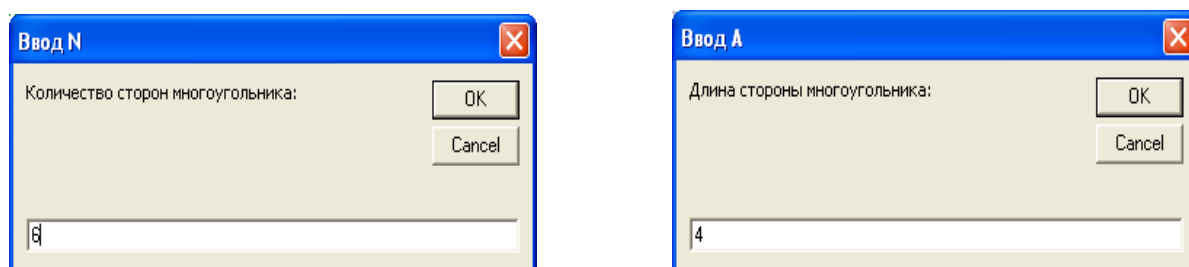


Рис. 2.4.1. Ввод данных в VBA-программу с помощью функции **InputBox()**.

InputBox (*Prompt* [, *Title*] [, *Default*] [, *XPos*] [, *YPos*] [, *HelpFile*, *Context*]), где

Prompt – любое строковое выражение. Аргумент *Prompt* является **единственным обязательным** аргументом для **InputBox()**, подсказывающим пользователю, что ему следует выполнить; все остальные – необязательные (об этом говорят квадратные скобки, в которые заключены эти аргументы вместе с разделяющими запятыми).

Title – это строка, используемая в качестве заголовка для окна ввода. Без этого аргумента в заголовке будет выведено наименование host-приложения, из которого запускается VBA-программа.

Default – любое строковое выражение, используемое как значение по умолчанию для пользовательского ввода (если это значение отсутствует, содержимое текстового поля так же отсутствует).

XPos, *YPos* – численные выражения, определяющие координаты левого верхнего угла окна ввода. *XPos* – горизонтальное расстояние от левого края активного окна; *YPos* – вертикальное расстояние от верхнего края активного окна.

HelpFile – строка, содержащая имя справочного файла Windows; обычно это – файл, который должен быть заранее создан с помощью Windows Help Compiler.

Аргумент *Context* – это численное выражение, указывающее на раздел в справочном файле, относящийся к отображаемому диалоговому окну.

В рассматриваемом примере использован **краткий** синтаксис функции **InputBox()**:

InputBox (*Prompt*, *Title*).

Для преобразования данных строкового типа, получаемых из текстового поля окна ввода, в числовой тип, используется функция **Val()**.

Val(S) – возвращает численное значение, соответствующее числу, представленному строкой *S*, которая должна содержать только цифры и одну десятичную точку, иначе VBA не сможет преобразовать её в число. В последнем случае функция **Val(S)** возвращает **0**.

Вывод результата для просмотра на экране можно выполнить с помощью операторов **Debug.Print**, позволяющих вывести результаты вычислений в окно **Immediate** (рис. 2.4.2) среды VBE в процессе выполнения программы. Для этого необходимо добавить в процедуру VBA операторы, вызывающие метод **Print** объекта **Debug**.

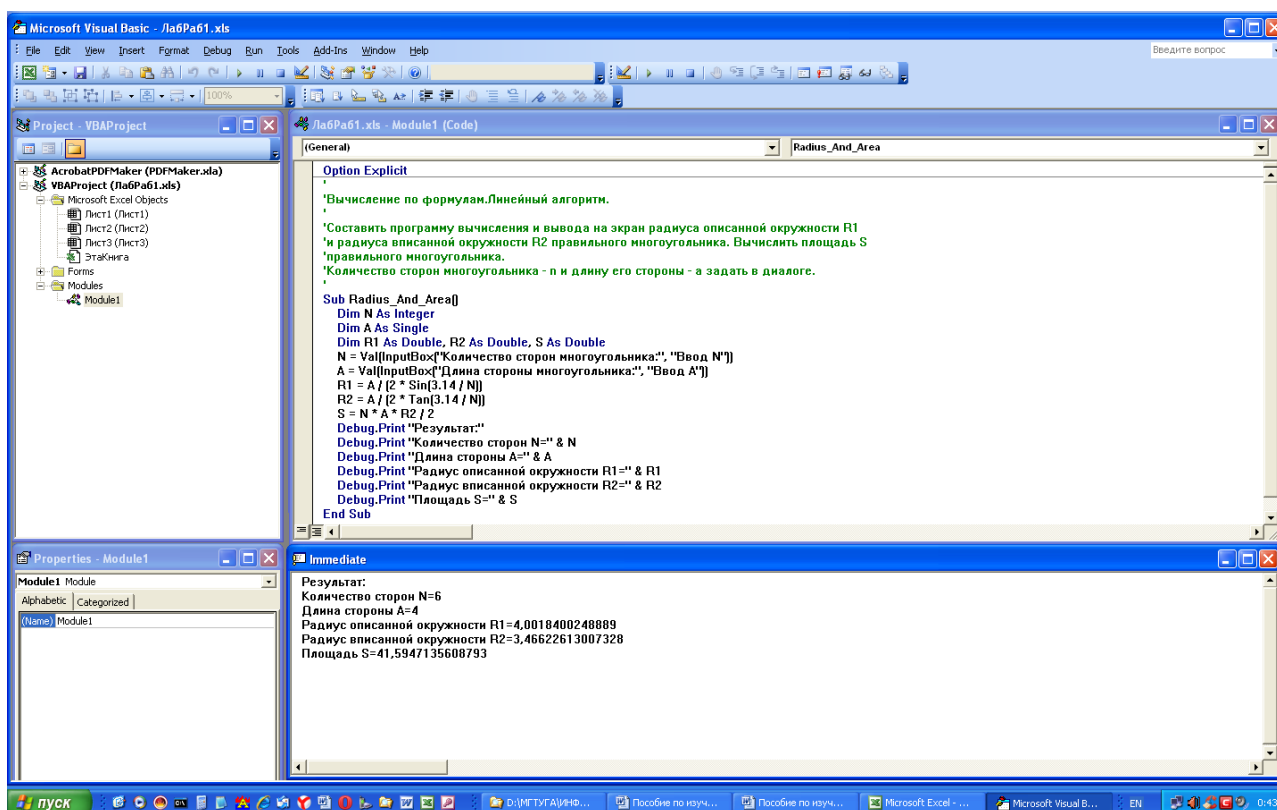


Рис. 2.4.2. Вывод результатов оператором **Debug.Print** в окно Immediate.

2.4.2. Задание № 7. Ветвящиеся алгоритмы

Написать VBA-программу и схему алгоритма вычисления и вывода на экран значений функции для исходных данных (данные выбрать самостоятельно), вводимых с клавиатуры компьютера в диалоговом режиме:

№	Функция	
1.	$F(x,y) =$	$\begin{cases} 3x+1, & \text{если } x < 0 \\ y + \cos(x), & \text{если } x = 0 \\ y - \sin(x), & \text{если } x > 0 \end{cases}$
2.	$F(x,y) =$	$\begin{cases} y \cdot x, & \text{если } x \leq -1,75 \\ 74, & \text{если } -1,75 < x < 0,28 \\ 3x + x - 6, & \text{если } x \geq 0,28 \end{cases}$
3.	$F(x,y) =$	$\begin{cases} x + y, & \text{если } x < y \\ 74x \cdot (y-9x) & \text{если } x \geq y \end{cases}$

4.	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } 2x + 3 > 0 \\ 74y, & \text{если } 2x + 3 = 0 \\ 3x + x - 6, & \text{если } 2x + 3 < 0 \end{array} \right.$
5.	$F(x,y) =$	$\left\{ \begin{array}{ll} \max(x,y+5), & \text{если } x > y \\ \min(x+1,y,3), & \text{если } x < y \\ x y, & \text{если } x = y \end{array} \right.$
6.	$F(x,y) =$	$\left\{ \begin{array}{ll} x + \sin(y), & \text{если } x - y > 0 \\ y - \cos(x), & \text{если } x - y < 0 \\ \operatorname{tg}(x) + 4y, & \text{если } x = y \end{array} \right.$
7.	$F(x,y) =$	$\left\{ \begin{array}{ll} 1, & \text{если } x < 1 \\ x + 5, & \text{если } 1 < x < 2 \\ 6y \cdot 2x + 4, & \text{если } 2 < x < 4 \\ x + \sin(y), & \text{если } 4 < x \end{array} \right.$
8.	$F(x,y) =$	$\left\{ \begin{array}{ll} x - y , & \text{если } x > y \\ y - x + 1, & \text{если } x = y \\ x^4 + \sin^3(y), & \text{если } x < y \end{array} \right.$
9.	$F(x,y) =$	$\left\{ \begin{array}{ll} x^y + y^x, & \text{если } x > y \\ x y, & \text{если } x < y \\ 2x - y , & \text{если } x = y \end{array} \right.$
10.	$F(x,y) =$	$\left\{ \begin{array}{ll} x(x^y - 1), & \text{если } x > y \\ x - \cos(xy), & \text{если } x < y \\ 7 + y, & \text{если } x = y \end{array} \right.$
11.	$F(x,y) =$	$\left\{ \begin{array}{ll} x y / (x+y), & \text{если } x > y \\ 6y \cdot 2x + 4, & \text{если } x < y \\ x + \sin(y), & \text{если } x = y \end{array} \right.$
12.	$F(x,y) =$	$\left\{ \begin{array}{ll} \min(x, 2y, 23), & \text{если } x > y \\ \max(2x, y), & \text{если } x < y \\ x + 1/y, & \text{если } x = y \end{array} \right.$
13.	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } x > y \\ (2x - 1) \cdot y, & \text{если } x < y \\ 578, & \text{если } x = y \end{array} \right.$
14.	$F(x,y) =$	$\left\{ \begin{array}{ll} x^2 + 1/y, & \text{если } x > y \\ 6y \cdot 2x + 4, & \text{если } x < y \\ x + \sin(y), & \text{если } x = y \end{array} \right.$

15.	$F(x,y) =$	$\left\{ \begin{array}{ll} \min(x, 2.5y), & \text{если } x > y \\ x - y , & \text{если } x = y \\ x \cdot y / (x+y), & \text{если } x < y \end{array} \right.$
16.	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x^3, & \text{если } x > y \\ 8y - 7x, & \text{если } x < y \\ \cos(x) + y, & \text{если } x = y \end{array} \right.$
17.	$F(x,y) =$	$\left\{ \begin{array}{ll} x^{y-1} - 9, & \text{если } x > y \\ 6y \cdot 2x + 41, & \text{если } x < y \\ 2x + \sin(y), & \text{если } x = y \end{array} \right.$
18.	$F(x,y) =$	$\left\{ \begin{array}{ll} \ln(x+y), & \text{если } x > y \\ \max(7x, 4y), & \text{если } x < y \\ 1/x + 1/y, & \text{если } x = y \end{array} \right.$
19.	$F(x,y) =$	$\left\{ \begin{array}{ll} 5 - y, & \text{если } x < y \\ \lg(x \cdot y), & \text{если } x = y \\ \min(x, 2y) , & \text{если } x > y \end{array} \right.$
20.	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } x > y \\ 6y \cdot 2x + 4, & \text{если } x < y \\ x + \sin(y), & \text{если } x = y \end{array} \right.$
21.	$F(x,y) =$	$\left\{ \begin{array}{ll} 1/x + 1/y, & \text{если } x - 0,5 > y \\ 6y \cdot 2x + 4, & \text{если } x < y + 0,5 \end{array} \right.$
22.	$F(x,y) =$	$\left\{ \begin{array}{ll} 8^{x/3} + y^{1/2}, & \text{если } x > y \\ \ln(x+1), & \text{если } x < y \\ y - 5x, & \text{если } x = y \end{array} \right.$
23.	$F(x,y) =$	$\left\{ \begin{array}{ll} 4^x - 2^{y-1} - 24, & \text{если } x > y \\ 6y \cdot 2x + 4, & \text{если } x < y \\ x + \sin(y), & \text{если } x = y \end{array} \right.$
24.	$F(x,y) =$	$\left\{ \begin{array}{ll} \min(x+1, 2y, 3), & \text{если } x > y \\ \max(x+5, y), & \text{если } x < y \\ (x/3)^y, & \text{если } x = y \end{array} \right.$
25.	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } 2x > y \\ 3x \cdot \sin(y), & \text{если } 2x < y \\ 5y \cdot \cos(x), & \text{если } 2x = y \end{array} \right.$

26.	$F(x,y) =$	$\left\{ \begin{array}{ll} 3x \cdot \sin(x) ^y, & \text{если } x > 6,5y \\ 6y \cdot 2x + 4, & \text{если } x < 6,5y \\ x + \sin(y), & \text{если } x = 6,5y \end{array} \right.$
27.	$F(x,y) =$	$\left\{ \begin{array}{ll} \arctg(x/2), & \text{если } x > y \\ \max(x+4,y), & \text{если } x < y \\ xy / (x+y)^{xy}, & \text{если } x = y \end{array} \right.$
28.	$F(x,y) =$	$\left\{ \begin{array}{ll} x^{2/3} + y^3 - 4xy, & \text{если } x > y \\ x+1, & \text{если } x < y \\ y - 5x, & \text{если } x = y \end{array} \right.$
29.	$F(x,y) =$	$\left\{ \begin{array}{ll} (x+y) ^{xy} - 4,2, & \text{если } x > y \\ \max(x,y - 5), & \text{если } x < y \\ \min(x+5,y^2,3), & \text{если } x = y \end{array} \right.$
30.	$F(x,y) =$	$\left\{ \begin{array}{ll} \arctg(x y), & \text{если } x > y \\ \max(y \cdot x, y+5), & \text{если } x < y \\ xy / (x+y)^{xy}, & \text{если } x = y \end{array} \right.$

Пример 7. Даны три точки – А,В,С на плоскости, на **различном** удалении от начала координат. Определить, какая из них расположена ближе к началу координат.

Option Explicit

'Вычисление по формулам. **Ветвящийся** алгоритм.

Sub Distance()

Dim xA As Single, xB As Single, xC As Single

Dim yA As Single, yB As Single, yC As Single

Dim DistA As Double

Dim DistB As Double

Dim DistC As Double

Dim strResultat As String

'Ввод в диалоговом окне InputBox

xA = Val(InputBox("Введите координаты точки А:", "Ввод xA"))

yA = Val(InputBox("Введите координаты точки А:", "Ввод yA"))

xB = Val(InputBox("Введите координаты точки В:", "Ввод xB"))

yB = Val(InputBox("Введите координаты точки В:", "Ввод yB"))

xC = Val(InputBox("Введите координаты точки С:", "Ввод xC"))

yC = Val(InputBox("Введите координаты точки С:", "Ввод yC"))

'Вычисление

DistA = Sqr(xA * xA + yA * yA)

```

DistB = Sqr(xB * xB + yB * yB)
DistC = Sqr(xC * xC + yC * yC)
If (DistA < DistB) And (DistA < DistC) Then
  strResultat = "Это точка А"
Elseif (DistB < DistA) And (DistB < DistC) Then
  strResultat = "Это точка В"
Else
  strResultat = "Это точка С"
End If
'Вывод в окно отладки
Debug.Print "Результат:"
Debug.Print "xA=" & xA, "yA=" & yA
Debug.Print "xB=" & xB, "yB=" & yB
Debug.Print "xC=" & xC, "yC=" & yC
Debug.Print strResultat
'Вывод в диалоговое окно MsgBox
MsgBox "Ближайшая точка: " & strResultat, , "Результат"
End Sub

```

Пусть точки **A**, **B** и **C** имеют координаты:

xA=1, yA=2;

xB=4, yB=2;

xC=5, yC=1.

После запуска процедуры, операторы ввода-вывода будут последовательно отображать на экране диалоговые окна `InputBox` и окно `MsgBox` (рис. 2.4.3).

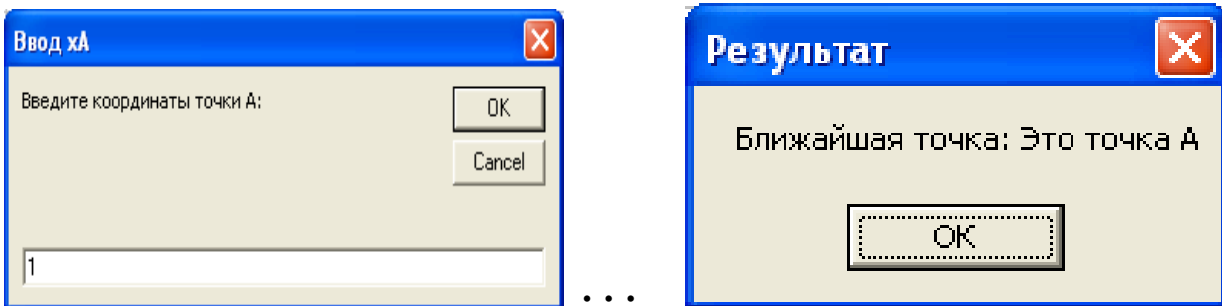


Рис. 2.4.3. Ввод-вывод с помощью диалоговых окон функций `InputBox ()` и `MsgBox()`

Режим **вывода** реализован (помимо использования окна отладки `Immediate`) с помощью функции `MsgBox()`, позволяющей выводить сообщения в специальное окно, в котором можно разместить определенный набор кнопок и информационный значок о типе сообщения. В зависимости от выбранной пользователем кнопки функция `MsgBox()` возвращает определенное значение, которое может быть присвоено числовой переменной в программе и использовано для анализа реакции пользователя на сообщение.

Функция `MsgBox()` имеет **полный** синтаксис:

`MsgBox (Prompt [, Buttons] [, Title] [, HelpFile, Context])`, где

Prompt – любое строковое выражение. Аргумент *Prompt* является **единственным обязательным** аргументом для MsgBox() и содержит сообщение, выводимое в диалоговом окне; все остальные – необязательные.

Аргументы *Title*, *HelpFile* и *Context* аналогичны соответствующим аргументам, описанным в синтаксисе функции InputBox().

Аргумент *Buttons* является численным выражением, которое задает количество и тип кнопок в диалоговом окне MsgBox, определяет кнопку по умолчанию. Кроме этого, с помощью аргумента Buttons можно указать, какие из стандартных значков Windows будут появляться в окне, сопровождая предупредительные сообщения и вопросы пользователю.

Перечень **аргументов-констант** функции MsgBox() и их назначение, а также возвращаемые функцией **значения-константы** приведены в соответствующих таблицах [3].

В рассматриваемом примере использован **краткий синтаксис** функции MsgBox():

MsgBox *Prompt* , , *Title*.

Кроме того, вывод результата можно наблюдать в окне отладки (Immediate):

Результат:

xA=1 yA=2

xB=4 yB=2

xC=5 yC=1

Это точка A

2.4.3. Задание № 8. Циклические алгоритмы

2.4.3.1. Задача № 8.1

8.1а. Написать VBA-программу и схему алгоритма вычисления функции для всех значений аргумента x (табулирование функции) на заданном интервале значений.

8.1б. Написать VBA-программу и схему алгоритма формирования массива значений функции для всех значений аргумента x_i . Массив $\{x_1, x_2, x_3, \dots, x_N\}$ задать с помощью датчика случайных чисел; N – задать в диалоге.

№	ВЫЧИСЛИТЬ ФУНКЦИЮ:	ИСХОДНЫЕ ДАННЫЕ ДЛЯ ЗАДАНИЙ 8.1а, 8.1б
1	$2\cos^3 x + 4e^{-x} + ax^4$	$-2 \leq x \leq 2, \Delta x=0.4, a=0.5$
2	$y^2 + x^2 + 5xy + 78$	$-2 \leq x \leq 4, \Delta x=0.5, y=3.2$
3	$y^x \cdot \arctg [(1 - x^3 / y)]$	$1 \leq x \leq 3, \Delta x=0.4, y=2.2$
4	$\arctg (y / x) + y^3 - x^2 + 1$	$0.4 \leq x \leq 2.6, \Delta x=0.6, y=3$
5	$ x ^{0,8} + \sin y - y^{-3} + \cos x$	$-1 \leq x \leq 2, \Delta x=0.3, y=1.2\pi$

6	$\sin y - y^{-3} + \ln \cos x + x^{0,8}$	$\pi \leq x \leq 2\pi, \Delta x=0.5, y=3.6$
7	$3,22y + 6,27 \operatorname{arctg} \left (1 - x^2 / y^2) \right ^{-1/2}$	$-1 \leq x \leq 2, \Delta x=0.4, y=1.4$
8	$\left 2x - y / \ln(x + y) + \sin(y) \right ^{-1/5}$	$4 \leq x \leq 8, \Delta x=0.2, y=2$
9	$x^3 + 1 / y + \log_3(7x)$	$-2 \leq x \leq 4, \Delta x=0.5, y=3.2$
10	$\left (x+1)^{1/2} - y^{1/3} \right \cdot \sin(x)$	$-\pi \leq x \leq \pi, \Delta x=0.5, y=3.6$
11	$x / [7bx^2 - ax^3 + \cos(3/4x)] - \sin(x)$	$-4 \leq x \leq 3, \Delta x=0.7, a=1.4, b=0.9$
12	$8^{2/x} - 2^{3x+3} + \operatorname{tg}(x/y)$	$0.1 \leq x \leq 5, \Delta x=0.6, y=3.6$
13	$[y - \sin(x)] / \log_9(xy) + \left (x-2y) \right ^{-1/3}$	$1 \leq x \leq 3, \Delta x=0.2, y=1.6$
14	$\left 2x \right ^{1/3} + 4xy - \sin x / (2x - y)$	$-2 \leq x \leq 3.14, \Delta x=0.3, y=1.2, b=3$
15	$5x + 2\sin^2 y - 4 / (x+7,2) + x^2$	$-2 \leq x \leq 3, \Delta x=0.5, y=0.2\pi, a=1.7$
16	$6,28x + 3,14y \ln \left \left(\frac{1-x}{1+x} \right) \right $	$-2 \leq x \leq 3, \Delta x=0.3, y=0.2\pi,$
17	$(3x + 1)^{3,6} \cdot (y + \cos x)^{-2} + \ln(xy)$	$1 \leq x \leq 3, \Delta x=0.2, y=1.2$
18	$\left (3x + yx - 6) \right ^{1/2} \cdot \ln(x + y)$	$0.1 \leq x \leq 2, \Delta x=0.2, y=-3$
19	$\left x + (y-9x)^2 (\operatorname{arctg} x) \right ^{-1/2}$	$-\pi \leq x \leq \pi, \Delta x=0.5, y=6$
20	$xy / (x+y) \cdot \left \operatorname{arctg}(1 - x^2 / y^2) \right ^{-1/2}$	$0.1 \leq x \leq 2, \Delta x=0.2, y=-2.2$
21	$7,45e^{xy} + 6,98 \sin(xy)$	$0.4 \leq x \leq 2, \Delta x=0.2, y=0.5$
22	$\log_3(7x^2) - \left \sin(x+y) / \cos(x) \right $	$-1.5 \leq x \leq 2, \Delta x=0.4, y=1.4$
23	$4^x - 2^{x-1} - \sin(3x / (x-1))$	$0.5\pi \leq x \leq \pi, \Delta x=0.5, y=1.5$
24	$\log_4 x^2 + \log_5(xy)^3 + \operatorname{tg}(x)$	$0.4 \leq x \leq 1.6, \Delta x=0.2, y=5$
25	$\sin^2(x+y) + \cos^2(xy)$	$0.5\pi \leq x \leq \pi, \Delta x=0.1, y=0.5\pi$
26	$\operatorname{tg}^2(x) + 4y + \log_3(7x^2)$	$0.2 \leq x \leq 2, \Delta x=0.2, y=0.5$

27	$y^2 + x^2 + 5xy + 78e^{\sin x}$	$4 \leq x \leq 8, \Delta x=0.5, y=2$
28	$\sqrt[3]{\left \frac{\sin(x+y)}{x(x+1)} + \frac{x - \cos(xy)}{7+y} \right }$	$0.5\pi \leq x \leq \pi, \Delta x=0.1, y=0.5\pi$
29	$(1/x)^3 + (1/y)^2 + \text{tg}(xy)$	$0.4 \leq x \leq 2.6, \Delta x=0.6, y=3$
30	$2 \sin^3(\text{arctg } x - \sin(x/2))$	$0.5\pi \leq x \leq \pi, \Delta x=0.1, y=1.5$

Пример 8.1а. Написать VBA-программу вычисления значений функции $F(X)$ для всех значений аргумента X , если:

$$F(X) = \text{SIN}(X) + \text{COS}(X),$$

при $-\pi \leq x \leq \pi$, шаг изменения аргумента $\Delta x = 0.6$

Option Explicit

'Вычисление функции на заданном интервале. Циклический алгоритм.

```
Sub Algorithm_81a()
Dim F As Double, X As Single
Const PI = 3.14
Debug.Print "Результат:"
Debug.Print "Табулирование функции без массива"
Debug.Print "X", , "F"
For X = -PI To PI Step 0.6
    F = Sin(X) + Cos(X)
    Debug.Print X, , F
Next X
End Sub
```

Результат:

Табулирование функции без массива

X	F
-3,14	-1,00159127990706
-2,54	-1,39039152898612
-1,94	-1,29348785872034
-1,34	-0,744731630697998
-0,7399999	6,41808020420419E-02
-0,1399999	0,850673032407508
0,4600002	1,34000067563249
1,06	1,36122749471902
1,66	0,90693835048925
2,26	0,135829523234772
2,86	-0,68272832034195

Пример 8.16. Написать VBA-программу формирования массива значений функции $F(X_i)$ для всех значений аргумента X_i из диапазона $-\pi \leq X_i \leq \pi$, если:

$$F(X_i) = \text{SIN}(X_i) + \text{COS}(X_i),$$

массив $\{X_1, X_2, X_3, \dots, X_N\}$ задать с помощью датчика случайных чисел, размерность массива N – задать в диалоге.

Option Explicit

'Вычисление функции от аргумента, заданного

'массивом случайных чисел. **Циклический** алгоритм.

Sub Algorithm_81b()

Dim I As Integer, N As Integer

Const PI = 3.14

N = Val(InputBox("Введите размерность массива", "Ввод N"))

ReDim X(N) As Single, F(N) As Double

Randomize

Debug.Print "Результат:"

Debug.Print "Табулирование функции с массивом X(" & N & "):"

Debug.Print "X(I)", "F(I)"

For I = 1 To N

X(I) = -PI+2*PI* Rnd()

F(I) = Sin(X(I)) + Cos(X(I))

Debug.Print X(I), F(I)

Next I

End Sub

Результат:

Табулирование функции с массивом X(10):

X(I)	F(I)
2,974517	-0,819775984561735
2,548906	-0,270851592575312
2,71225	-0,492966683205004
2,030184	0,452924783121543
2,689984	-0,463332224563144
-1,802718	-1,2030748786946
1,552797	1,01783664489358
-0,5184772	0,373016773068505
-1,412968	-0,830397362806632
-2,917056	-1,19755216520793

2.4.3.2. Задача № 8.2

8.2а. Вычислить сумму ряда для заданного количества слагаемых N . Значение переменной задать в диалоговом режиме самостоятельно. На экран вывести значение суммы ряда и количество слагаемых.

8.2б. Вычислить сумму ряда с заданной точностью ϵ . Значение точности задать в диалоговом режиме самостоятельно. На экран вывести значение суммы ряда и количество итераций.

Сравнить с результатом, полученным в задании 8.2а, для одного и того же числа итераций.

Примечание 5. В тех вариантах, где **общий член ряда** не задан, его следует вывести самостоятельно.

$$1. S = 1 - \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3} - \frac{1}{1 \cdot 2 \cdot 3 \cdot 4} + \frac{1}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5} - \dots + (-1)^n \frac{1}{(N+1)!}$$

$$2. S = 1 - \frac{4}{7} + \frac{4}{11} - \frac{4}{15} + \frac{4}{19} - \frac{4}{23} + \dots$$

$$3. S = 1 + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \dots + \frac{1}{2N+1}$$

$$4. S = 1 - \frac{1}{2} + \frac{1}{4} - \frac{1}{8} + \frac{1}{16} - \dots + (-1)^n \frac{1}{2^n}$$

$$5. S = \frac{5}{3} + \frac{5}{6} - \frac{5}{9} + \frac{5}{12} - \frac{5}{15} + \dots$$

$$6. S = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots + (-1)^n \frac{1}{2N+1}$$

$$7. S = 1 + \frac{1}{2} - \frac{1}{5} + \frac{1}{8} - \frac{1}{11} + \dots - (-1)^n \frac{1}{2+3N}$$

$$8. S = -\frac{7}{1} + \frac{7}{5} - \frac{7}{9} + \frac{7}{13} - \frac{7}{17} + \dots$$

$$9. S = -\frac{2}{4} + \frac{2}{7} - \frac{2}{10} + \frac{2}{13} - \frac{2}{16} + \dots$$

$$10. S = 1 + \frac{1}{2} + \frac{1}{5} + \frac{1}{8} + \frac{1}{11} + \dots + \frac{1}{2+3N}$$

$$11. S = -\frac{1}{1} + \frac{1}{5} + \frac{1}{9} + \frac{1}{13} + \frac{1}{17} + \dots$$

$$12. S = -\frac{3}{2 \cdot 4} + \frac{3}{2 \cdot 7} - \frac{3}{2 \cdot 10} + \frac{3}{2 \cdot 13} - \frac{3}{2 \cdot 16} + \dots$$

$$13. S = 1 + \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3} + \frac{1}{1 \cdot 2 \cdot 3 \cdot 4} + \frac{1}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5} + \dots + \frac{1}{N!}$$

$$14. S = 4 + \frac{4}{4} + \frac{4}{9} + \frac{4}{16} + \frac{4}{25} + \frac{4}{36} + \dots + \frac{4}{N^2}$$

$$15. S = 1 + \frac{7}{2} + \frac{7}{4} + \frac{7}{8} + \frac{7}{16} + \dots + \frac{7}{2^n}$$

$$16. S = \frac{3}{2} + \frac{3}{5} - \frac{3}{8} + \frac{3}{11} - \frac{3}{14} + \dots$$

$$17. S = \frac{2}{2} + \frac{2}{5} + \frac{2}{8} + \frac{2}{11} + \dots + \frac{2}{2 + 3N}$$

$$18. S = \frac{13}{1} + \frac{13}{5} + \frac{13}{9} + \frac{13}{13} + \frac{13}{17} + \dots$$

$$19. S = \frac{1}{3} + \frac{1}{8} + \frac{1}{15} + \frac{1}{24} + \dots + \frac{1}{(N-1)(N+1)}$$

$$20. S = 1 + \frac{1}{1 \cdot 2} + \frac{2}{1 \cdot 2 \cdot 3} + \frac{3}{1 \cdot 2 \cdot 3 \cdot 4} + \frac{4}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5} + \dots + \frac{(N-1)}{N!}$$

$$21. S = 1 + 0.7 + 0.7^2 + 0.7^3 + 0.7^4 + \dots$$

$$22. S = 1 + \frac{2}{1} + \frac{3}{1 \cdot 2} + \frac{4}{1 \cdot 2 \cdot 3} + \dots + \frac{N}{(N-1)!}$$

$$23. S = 1 + \frac{2}{1 \cdot 2} + \frac{3}{1 \cdot 2 \cdot 3} + \frac{4}{1 \cdot 2 \cdot 3 \cdot 4} + \dots + \frac{N}{N!}$$

$$24. S = \frac{1}{3} - \frac{1}{9} + \frac{1}{27} - \frac{1}{81} + \dots - (-1)^{n-1} \frac{1}{3^n}$$

$$25. S = 1 + \frac{1}{2} - \frac{1}{7} + \frac{1}{12} - \frac{1}{17} + \dots - (-1)^n \frac{1}{2 + 5N}$$

$$26. S = \frac{5}{1} + \frac{5}{5} + \frac{5}{9} + \frac{5}{13} + \frac{5}{17} + \dots$$

$$27. S = -\frac{2}{4} + \frac{2}{9} - \frac{2}{14} + \frac{2}{19} - \frac{2}{24} + \dots$$

$$28. S = 3 + \frac{3}{1 \cdot 2} + \frac{3}{1 \cdot 2 \cdot 3} + \frac{3}{1 \cdot 2 \cdot 3 \cdot 4} + \frac{3}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5} + \dots + \frac{3}{N!}$$

$$29. S = 1 + \frac{3}{6} + \frac{3}{11} + \frac{3}{16} + \frac{3}{21} + \frac{3}{26} + \dots$$

$$30. S = 1 + \frac{1}{3} + \frac{1}{6} + \frac{1}{9} + \frac{1}{12} + \dots + \frac{1}{3N}$$

Пример 8.2а. Вычислить сумму n слагаемых ряда:

$$S = 1 + \frac{1}{3} + \frac{1}{6} + \frac{1}{9} + \dots + \frac{1}{3n} = 1 + \sum_{i=1}^n \frac{1}{3i}$$

Option Explicit

'Вычисление суммы ряда. Цикл с заданным числом повторений.

Sub Algorithm_82a()

Dim I As Integer, N As Integer

Dim S As Double

N = Val(InputBox("Число слагаемых ряда:", "Ввод N"))

S = 1

For I = 1 To N

S = S + 1 / (3 * I)

Next I

Debug.Print "Результат:"

Debug.Print "Сумма ряда:", S, "Число слагаемых ряда:", N

End Sub

Результат:

Сумма ряда: 3,89643123024042

Число слагаемых ряда:

3334

Пример 8.2б. Вычислить сумму ряда с заданной точностью e :

$$S = 1 + \frac{1}{3} + \frac{1}{6} + \frac{1}{9} + \dots + \frac{1}{3n} = 1 + \sum_{i=1}^n \frac{1}{3i}$$

Option Explicit

'Вычисление с заданной точностью суммы ряда. Цикл с неизвестным числом повторений.

Sub Algorithm_82b()

Dim I As Integer

Dim S As Double, A As Double ' A - общий член ряда

Dim e As Double

```

e = Val(TextBox("Точность вычисления суммы ряда:", "Ввод e"))
S = 1: I = 1
Do
  A = 1 / (3 * I)
  S = S + A
  I = I + 1
Loop While Abs(A) >= e
Debug.Print "Результат:"
Debug.Print "Точность вычисления:", e
Debug.Print "Сумма ряда:", S, "Число итераций:", I - 1
End Sub

```

Результат:

Точность вычисления: 0,0001

Сумма ряда: 3,89643123024042 Число итераций: 3334

2.4.4. Задание № 9. Алгоритмы смешанного типа

Написать VBA-программу вычисления функции и схему алгоритма смешанного типа. На экран вывести таблицу значений аргумента и функции:

№	Функция		Аргумент, параметры
1	$F(x,y) = \left\{ \begin{array}{l} 3x + x - 6, \\ 74y, \\ y^{x+1} - 5x, \end{array} \right.$	$\left. \begin{array}{l} \text{если } x < -0.9 \\ \text{если } -0.9 \leq x \leq 0.9 \\ \text{если } x > 0.9 \end{array} \right\}$	$-3 \leq x \leq 2,$ $\Delta x = 0.3, y = 8.1$
2	$F(x,y) = \left\{ \begin{array}{l} 3x+1, \\ y + \cos(x), \\ y - \sin(x), \end{array} \right.$	$\left. \begin{array}{l} \text{если } x < -2 \\ \text{если } -2 \leq x \leq 2 \\ \text{если } x > 2 \end{array} \right\}$	$-3 \leq x \leq 3,$ $\Delta x = 0.5, y = 5.8$
3	$F(x,y) = \left\{ \begin{array}{l} x + y, \\ 74x \cdot (y-9x) \end{array} \right.$	$\left. \begin{array}{l} \text{если } x < 0 \\ \text{в противном случае} \end{array} \right\}$	$-3 \leq x \leq 2,$ $\Delta x = 0.5, y = 1.9$

4	$F(x,y) =$	$\left\{ \begin{array}{ll} xy+5, & \text{если } x < -0.7 \\ x+5y, & \text{если } -0.7 \leq x \leq 0.7 \\ x y, & \text{если } x > 0.7 \end{array} \right.$	$-2 \leq x \leq 2,$ $\Delta x = 0.3, y = 2.4$
5	$F(x,y) =$	$\left\{ \begin{array}{ll} x + \sin(y), & \text{если } x < -1.5 \\ y - \cos(x), & \text{если } -1.5 \leq x \leq 2 \\ \operatorname{tg}(x) + 4y, & \text{если } x > 2 \end{array} \right.$	$-3.5 \leq x \leq 4,$ $\Delta x = 0.7, y = 1.6$
6	$F(x,y) =$	$\left\{ \begin{array}{ll} y \cdot x, & \text{если } x \leq -1,75 \\ 74, & \text{если } -1,75 < x < 0,28 \\ 3x + x - 6, & \text{если } x \geq 0,28 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.5, y = 2.2$
7	$F(x,y) =$	$\left\{ \begin{array}{ll} x^y + y^x, & \text{если } x < 0.9 \\ x y, & \text{если } 0.9 \leq x \leq 1.5 \\ 2x - y , & \text{если } x > 1.5 \end{array} \right.$	$0.3 \leq x \leq 2,$ $\Delta x = 0.1, y = 2.1$
8	$F(x,y) =$	$\left\{ \begin{array}{ll} x(x^y - 1), & \text{если } x < 2 \\ x - \cos(xy), & \text{если } 2 \leq x \leq 4 \\ 7 + y, & \text{если } x > 4 \end{array} \right.$	$0.5 \leq x \leq 5,$ $\Delta x = 0.3, y = 8$
9	$F(x,y) =$	$\left\{ \begin{array}{ll} x - y , & \text{если } x < 0 \\ x^4 + \sin^3(y) & \text{если } 0 \leq x \end{array} \right.$	$-2 \leq x \leq 2,$ $\Delta x = 0.5, y = 1.3$
10	$F(x,y) =$	$\left\{ \begin{array}{ll} x y / (x+y), & \text{если } x < -1 \\ 6y \cdot 2x + 4, & \text{если } -1 \leq x \leq 1 \\ x + \sin(y), & \text{если } x > 1 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.3, y = 1.6$
11	$F(x,y) =$	$\left\{ \begin{array}{ll} (x - 2y - 23), & \text{если } x < -1 \\ 2x + y , & \text{если } -1 \leq x \leq 2 \\ x + 1/y, & \text{если } x > 2 \end{array} \right.$	$-3 \leq x \leq 5,$ $\Delta x = 0.8, y = -18$
12	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } x < 0 \\ (2x - 1) \cdot y, & \text{если } 0 \leq x \leq 2 \\ 578, & \text{если } x > 2 \end{array} \right.$	$-2 \leq x \leq 5,$ $\Delta x = 0.6, y = 5$
13	$F(x,y) =$	$\left\{ \begin{array}{ll} 1, & \text{если } x < -1 \\ x + 5, & \text{если } -1 \leq x \leq 1 \\ 6y \cdot 2x + 4, & \text{если } 1 < x \leq 2 \\ x + \sin(y), & \text{если } x > 2 \end{array} \right.$	$-3 \leq x \leq 3,$ $\Delta x = 0.3, y = 1.6$
14	$F(x,y) =$	$\left\{ \begin{array}{ll} x^2 + 1/y, & \text{если } x < -2 \\ 6y \cdot 2x + 4, & \text{если } -2 \leq x \leq 3 \\ x + \sin(y), & \text{если } x > 3 \end{array} \right.$	$-5 \leq x \leq 5,$ $\Delta x = 0.5, y = 3.1$

15	$F(x,y) =$	$\left\{ \begin{array}{l} (x - 2.5y), \quad \text{если } x < -1 \\ x^4 + \sin^3(y) \quad \text{если } -1 \leq x \leq 1 \\ x \cdot y / (x+y), \quad \text{если } x > 1 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.3, y = 1.6$
16	$F(x,y) =$	$\left\{ \begin{array}{l} y^{x+1} - 5x^3, \quad \text{если } x < -1 \\ 8y - 7x, \quad \text{если } -1 \leq x \leq 1 \\ \cos(x) + y, \quad \text{если } x > 1 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.3, y = 1$
17	$F(x,y) =$	$\left\{ \begin{array}{l} x^{y-1} - 9, \quad \text{если } x < 5 \\ 6y \cdot 2x + 41, \quad \text{если } 5 \leq x \leq 7 \\ 2x + \sin(y), \quad \text{если } x > 7 \end{array} \right.$	$3 \leq x \leq 10,$ $\Delta x = 0.9, y = 3.14$
18	$F(x,y) =$	$\left\{ \begin{array}{l} \ln(x+y), \quad \text{если } x < 1.1 \\ (7x - 4y), \quad \text{если } 1.1 \leq x \leq 1.6 \\ 1/x + 1/y, \quad \text{если } x > 1.6 \end{array} \right.$	$0.3 \leq x \leq 2,$ $\Delta x = 0.2, y = 1.1$
19	$F(x,y) =$	$\left\{ \begin{array}{l} 5 - y, \quad \text{если } x < -1 \\ \lg (x \cdot y) , \quad \text{если } -1 \leq x \leq -0.1 \\ (x + 2y) , \quad \text{если } x > -0.1 \end{array} \right.$	$-3 \leq x \leq 3,$ $\Delta x = 0.3, y = -2$
20	$F(x,y) =$	$\left\{ \begin{array}{l} y^{x+1} - 5x, \quad \text{если } x < -1.2 \\ 6y \cdot 2x + 4, \quad \text{если } -1.2 \leq x \leq 1 \\ x + \sin(y), \quad \text{если } x > 1 \end{array} \right.$	$-2 \leq x \leq 2,$ $\Delta x = 0.4, y = 3.14$
21	$F(x,y) =$	$\left\{ \begin{array}{l} 1/x + 1/y, \quad \text{если } x < 0 \\ 6y \cdot 2x + 4, \quad \text{если } x \geq 0 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.5, y = 2$
22	$F(x,y) =$	$\left\{ \begin{array}{l} 8^{x/3} + y^{1/2}, \quad \text{если } x < -5 \\ \ln(x+6), \quad \text{если } -5 \leq x \leq 2 \\ y - 5x, \quad \text{если } x > 2 \end{array} \right.$	$-9 \leq x \leq 5,$ $\Delta x = 1, y = 4$
23	$F(x,y) =$	$\left\{ \begin{array}{l} 4^x - 2^{y-1} - 24, \quad \text{если } x < -1 \\ 6y \cdot 2x + 4, \quad \text{если } -1 \leq x \leq 1 \\ x + \sin(y), \quad \text{если } x > 1 \end{array} \right.$	$-3 \leq x \leq 2,$ $\Delta x = 0.3, y = 3.1$
24	$F(x,y) =$	$\left\{ \begin{array}{l} \operatorname{arctg}(x \cdot y), \quad \text{если } x < -1.2 \\ (y^2 x + 5), \quad \text{если } -1.2 \leq x \leq 1.2 \\ xy / (x+y)^{xy}, \quad \text{если } x > 1.2 \end{array} \right.$	$-2 \leq x \leq 2,$ $\Delta x = 0.4, y = 0.7$
25	$F(x,y) =$	$\left\{ \begin{array}{l} (x + 1.2y + 3), \quad \text{если } x < -1.5 \\ (x + 5y), \quad \text{если } -1.5 \leq x \leq 1 \\ (x/3)^y, \quad \text{если } x > 1 \end{array} \right.$	$-1 \leq x \leq 2,$ $\Delta x = 0.1, y = 2$

26	$F(x,y) =$	$\left\{ \begin{array}{ll} y^{x+1} - 5x, & \text{если } x < -2 \\ 3x \cdot \sin(y), & \text{если } -2 \leq x \leq 2 \\ 5y \cdot \cos(x), & \text{если } x > 2 \end{array} \right.$	$-3 \leq x \leq 3,$ $\Delta x = 0.6, y = 8.1$
27	$F(x,y) =$	$\left\{ \begin{array}{ll} 3x \cdot \sin(x) ^y, & \text{если } x < 1.5 \\ 6y \cdot 2x + 4, & \text{если } 1.5 \leq x \leq 3 \\ x + \sin(y), & \text{если } x > 3 \end{array} \right.$	$0.3 \leq x \leq 4,$ $\Delta x = 0.3, y = 1.57$
28	$F(x,y) =$	$\left\{ \begin{array}{ll} \arctg(x/2), & \text{если } x < 3 \\ (x+4y), & \text{если } 3 \leq x \leq 6 \\ xy / (x+y)^{xy}, & \text{если } x > 6 \end{array} \right.$	$2 \leq x \leq 8,$ $\Delta x = 0.8, y = 3$
29	$F(x,y) =$	$\left\{ \begin{array}{ll} x^{2/3} + y^3 - 4xy, & \text{если } x < 2 \\ x+1, & \text{если } 2 \leq x \leq 3 \\ y - 5x, & \text{если } x > 3 \end{array} \right.$	$0.5 \leq x \leq 4,$ $\Delta x = 0.5, y = 2.1$
30	$F(x,y) =$	$\left\{ \begin{array}{ll} (x+y) ^{xy} - 4,2, & \text{если } x < -1 \\ (x^2 y - 5), & \text{если } -1 \leq x \leq 5 \\ (x+5+y^2-3), & \text{если } x > 5 \end{array} \right.$	$-3 \leq x \leq 9,$ $\Delta x = 0.8, y = 8.1$

Пример 9. Составить VBA-программу вычисления и печати всех значений функции $F(x)$ для всех значений аргумента x , если :

$F(x) =$	$\left\{ \begin{array}{ll} \sin x & , \text{если } x \leq -0.5\pi \\ \cos x & , \text{если } -0.5\pi < x < 0.5\pi \\ \operatorname{tg} x & , \text{если } x \geq 0.5\pi \end{array} \right.$	при $-\pi \leq x \leq \pi$, шаг изменения аргумента: $\Delta x = 0.1$
----------	--	---

Option Explicit

'Вычисление интервальной функции. Алгоритм **смешанного** типа.

Sub Algorithm_9()

Dim X As Single, dX As Single

Dim F As Double

Const PI = 3.14: X = -PI

dX = Val(InputBox("Значение шага изменения аргумента:", "Ввод dX"))

Debug.Print "Результат:"

Debug.Print "Для шага dX=" & dX

10 If X <= -0.5 * PI Then

F = Sin(X)

Elseif X >= 0.5 * PI Then

F = Tan(X)

```

Else
  F = Cos(X)
End If
'Вывод в окно отладки
Debug.Print "x=" & X, "F=" & F
X = X + dX
If X <= PI Then GoTo 10
End Sub

```

Результат:

Для шага dX=0,6

x=-3,14	F=-1,59254801244519E-03
x=-2,54	F=-0,565956261898413
x=-1,94	F=-0,932615036392286
x=-1,34	F=0,228752891363307
x=-0,7399999	F=0,738468632680456
x=-0,1399999	F=0,990216014843644
x=0,4600002	F=0,896052427667499
x=1,06	F=0,488871923791346
x=1,66	F=-11,1805245186337
x=2,26	F=-1,21359429471794
x=2,86	F=-0,289279318700279

2.5. Интерактивные приложения

Рассмотрим процесс создания простейшего пользовательского приложения, автоматизирующего решение некоторой задачи, связанной, например, с вычислением факториала натурального числа.

Разработку приложения можно представить в виде двух последовательно выполняемых этапов:

- разработку средств, реализующих функциональность приложения (процедурный этап разработки);
- разработку пользовательского интерфейса (объектный этап разработки).

2.5.1. Процедуры

На данном этапе следует отладить процедуру, выполняющую основную функцию приложения. Отладка производится в окне модуля (рис. 2.5.1). Ниже приведен код процедуры **Sub Factorial()**. Пока еще форма не разработана, ввод исходных можно выполнять обычным способом – с помощью диалогового окна **InputBox**, а вывод – в окно отладки **Immediate**. После окончания отладки процедуры, строки ввода-вывода можно трансформировать в комментарии, а вместо их написать инструкции ввода-вывода с помощью полей интерфейсной формы, которую, к этому моменту, следует разработать.

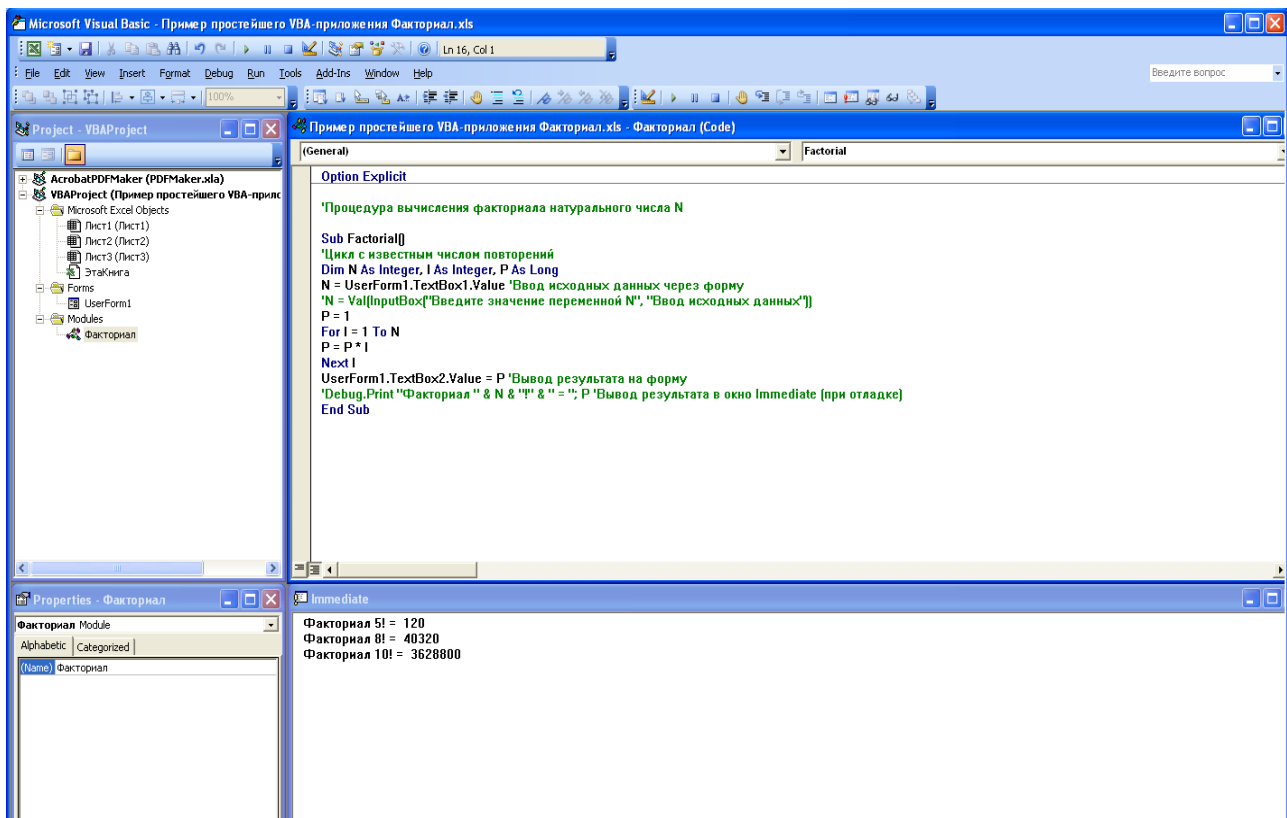


Рис. 2.5.1. Среда редактора VBE. Открыты окна:

- Окно Диспетчера проектов
- Окно Свойств объектов
- Окно Модуля процедуры
- Окно Immediate

Option Explicit

'Процедура вычисления факториала натурального числа N

Sub Factorial()

'Цикл с известным числом повторений

Dim N As Integer, I As Integer, P As Long

N = UserForm1.TextBox1.Value 'Ввод исходных данных на форму

'N = Val(InputBox("Введите значение переменной N", "Ввод исходных данных")) 'Ввод в
'диалоговом окне **InputBox** (при отладке процедуры)

P = 1

For I = 1 To N

P = P * I

Next I

UserForm1.TextBox2.Value = P 'Вывод результата на форму

'Debug.Print "Факториал " & N & "! " & " = "; P 'Вывод результата в окно Immediate (при
'отладке процедуры)

End Sub

Отладочная печать в окне Immediate

Факториал 5! = 120

Факториал 8! = 40320

Факториал 10! = 3628800

2.5.2. Интерфейсы

Интерфейсная форма с объектами управления показана на рис.2.5.2. в окне Конструктора форм. VBA-коды событийных процедур отображаются в окне Модуля формы. Вывод результата в окне Immediate можно использовать для сопоставления с выводом его в поле формы для подтверждения корректной работы процедуры и интерфейса.

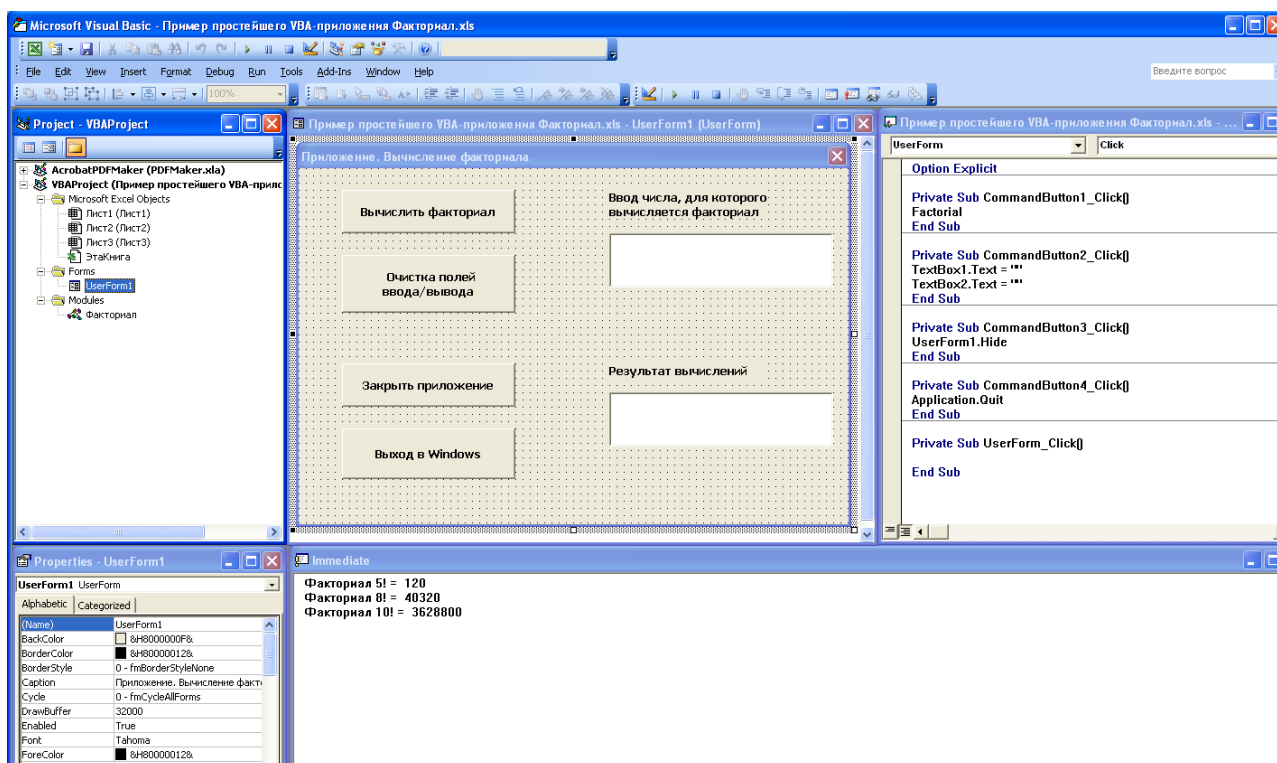


Рис. 2.5.2. Среда VBE. Открыты окна:

- Диспетчера проектов
- Конструктора форм
- Отладочной печати
- Свойств объектов
- Модуля формы

На форме расположены четыре объекта класса **CommandButton**, два объекта класса **TextBox** и два объекта класса **Label**. Приведенные ниже VBA-коды являются событийными процедурами, связанными с объектами класса **CommandButton**.

Option Explicit

```
Private Sub CommandButton1_Click() 'Вычислить факториал
    Factorial
End Sub
```

```
Private Sub CommandButton2_Click() 'Очистка полей ввода/вывода
    TextBox1.Text = ""
    TextBox2.Text = ""
End Sub
```

```
Private Sub CommandButton3_Click() 'Закреть приложение
UserForm1.Hide
End Sub
```

```
Private Sub CommandButton4_Click() 'Выход в Windows
Application.Quit
End Sub
```

Запуск пользовательского приложения осуществляется в открытом окне host-приложения, В рассматриваемом случае, это – Excel (рис. 2.5.3). VBA-код процедуры старта приложения состоит из единственной инструкции, вызывающей метод **Show** для объекта **UserForm1**. Заметьте, что эта процедура связана с объектом **CommandButton1** ("Запуск приложения"), расположенным на рабочем листе Excel – Листе1, и ничего общего не имеет с объектом **CommandButton1** ("Вычислить факториал"), расположенным на форме **UserForm1**.

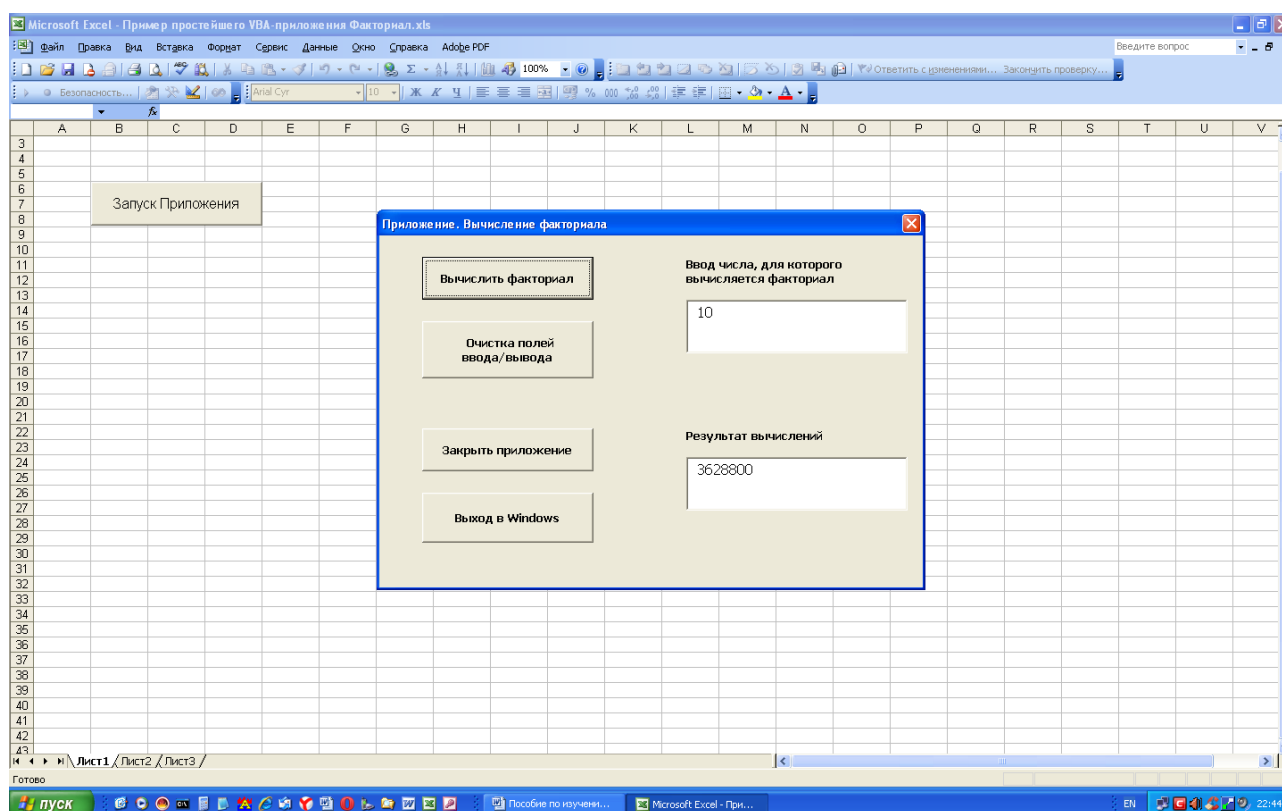


Рис. 2.5.3. Host-приложение Excel. Открыты окна:

- Рабочего Листа1 с кнопкой запуска пользовательского приложения
- Интерактивного графического интерфейса (появляется после запуска)

```
Sub CommandButton1_Click() 'Кнопка старта приложения на Листе1 Excel
UserForm1.Show
End Sub
```

2.6. Вопросы для самопроверки при подготовке к аттестации

1. Свойства и способы описания алгоритмов.
2. Схема алгоритма. Примеры.
3. Типы алгоритмических структур. Примеры.
4. Процесс разработки программ. Методы проектирования.
5. Объектно-ориентированный подход (ООП). Базовые понятия.
6. Объект. Свойства и методы. Примеры.
7. События. Классы объектов.
8. Встроенные типы данных VBA. Примеры.
9. Допустимые имена (идентификаторы).
10. Используемые в VBA арифметические операторы.
11. Операторы конкатенации.
12. Операторы сравнения.
13. Используемые в VBA логические операторы.
14. Приоритеты операций в выражениях.
15. Правила записи и использования оператора присваивания.
16. Правила оформления кода.
17. Использование окна ввода.
18. Использование окна сообщений.
19. Ввод данных с рабочего листа и вывод результатов в ячейки листа Excel.
20. Структуры хранения данных VBA. Переменные. Константы. Массивы. Примеры.
21. Операторы управления. Оператор ветвления. Оператор цикла с заданным количеством повторений. Оператор цикла по условию. Вложенные циклы. Примеры.
22. Функции ввода-вывода. Примеры
23. Проект. Модуль. Структура VBA-программы.
24. Операторы организации подпрограмм. Подпрограмма-процедура Sub (). Подпрограмма-функция Function (). Примеры.
25. С какой целью применяют подпрограммы?
26. Событийные процедуры. Примеры.
27. Использование функций для работы со строками. Примеры.
28. В чем отличие именованных от позиционных аргументов процедур?
29. Формальные и фактические параметры подпрограммы.
30. Раннее окончание процедур и функций.
31. Использование свойств объектов VBA.
32. Использование методов объектов VBA.
33. Редактор VBE. Назначение окна проекта. Использование окна свойств. Использование окна модуля. Работа с формами. Задание параметров редактора VBE.
34. Макросы VBA. Технология создания макроса. Исходный код макроса
35. Что определяет директива Option Explicit?