

# **ИНФОРМАТИКА**

## **ПРОГРАММИРОВАНИЕ НА VBA. ЛИНЕЙНЫЕ, РАЗВЕТВЛЯЮЩИЕСЯ И ЦИКЛИЧЕСКИЕ ПРОЦЕССЫ**

*Методические указания к курсовой работе  
для студентов бакалавриата направлений  
13.03.02, 20.03.01, 27.03.01*

**САНКТ-ПЕТЕРБУРГ  
2023**

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
Санкт-Петербургский горный университет

Кафедра информатики и компьютерных технологий

# ИНФОРМАТИКА

ПРОГРАММИРОВАНИЕ НА VBA. ЛИНЕЙНЫЕ,  
РАЗВЕТВЛЯЮЩИЕСЯ И ЦИКЛИЧЕСКИЕ ПРОЦЕССЫ

*Методические указания к курсовой работе  
для студентов бакалавриата направлений  
13.03.02, 20.03.01, 27.03.01*

САНКТ-ПЕТЕРБУРГ  
2023

УДК 004.432.2 (073)

**ИНФОРМАТИКА. Программирование на VBA. Линейные, разветвляющиеся и циклические процессы:** Методические указания к практическим занятиям / Санкт-Петербургский горный университет. Сост.: *С.Ю. Кротова, А.Е. Ильин, А.В. Чиргин*. СПб, 2023. 41 с.

В методическом пособии представлен теоретический материал и практические задания для выполнения практических занятий. Представлены примеры решения задач по программированию линейных, разветвляющихся процессов, а так же по работе с циклическими процессами представленными табулированием функций и обработкой массивов.

Методические указания предназначены для студентов специальностей 13.03.02 «Электроэнергетика и электротехника», 20.03.01 «Техносферная безопасность», 27.03.01 «Стандартизация и метрология».

Научный редактор доц. *Е.Н Овчинникова*

Рецензент к.т.н. *К.В. Столяров* (Корпорация «Телум Инк»)

© Санкт-Петербургский  
горный университет, 2023

## Введение

VBA (Visual Basic for Applications) — это язык программирования, встроенный во множество отдельных программ и прикладных пакетов — от приложений Microsoft Office (включая Microsoft Project и Microsoft Visio) и до таких мощных пакетов, как AutoCAD, CorelDraw и Adobe Creative Suite, не говоря уже о многочисленных специализированных приложениях, предназначенных для управления производственными процессами, учета финансовых ресурсов или информационной поддержки клиентов.

Он является наиболее удобным при необходимости автоматизировать обработку данных в MS Excel или MS Word. Электронные таблицы MS Excel разрабатывались для упрощения обработки больших объемов числовых данных, для выполнения рутинных операций и наглядного представления результатов вычислений. Даже неподготовленный пользователь может, пользуясь простыми формулами, выполнять бухгалтерские, статистические операции.

Используя особенности продуктов MS Office, при помощи VBA, можно автоматизировать решение практически любой задачи, связанной с этим пакетом программ. VBA позволяет максимально использовать готовые интерфейсные возможности Windows с максимально доступным интерпретатором. Идеология этого языка заключается в том, что основными при реализации процедур автоматизации является не программа, а документ, в котором хранится программа, хотя программы могут храниться и отдельно от документа.

В данном методическом пособии представлены примеры заданий по программированию линейных, разветвляющихся и циклических процессов, а так же задания для индивидуального решения по каждой теме.

## 1. Программирование линейного процесса

### 1.1 Функции ввода и вывода в диалоговые окна VBA

#### 1) Окно ввода

Для организации ввода данных в диалоговом режиме используется функция `InputBox`, в результате работы которой появляется диалоговое окно (Рис.1.1). Функция имеет следующий синтаксис:

**`InputBox(Сообщение, Заголовок, Текст по умолчанию)`**,

где

*Сообщение* – текст, максимальная длина 1024 символа.

Перенос текста – клавиша Enter;

*Заголовок* – выражение, которое отображается в верхней строке окна. Если этот аргумент опущен, в строке будет имя приложения, например, MS Excel;

*Текст по умолчанию* – выражение, отображаемое в поле ввода, если пользователь не введет другую строку. Если аргумент опущен, поле ввода отображается пустым.

Например:

**Результат = `InputBox(“Введите число от 1 до 3”, “Пример”, “1”)`**

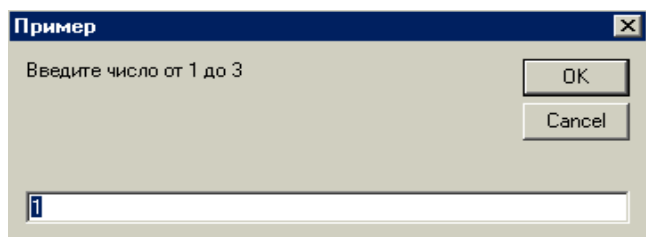


Рис. 1.1 Диалоговое окно функции `InputBox`

#### 2) Окно сообщений

Для организации вывода в диалоговом режиме используют функцию `MsgBox`, которая помимо вывода содержимого сообщения или результата вычислений, устанавливает режим ожидания

нажатия кнопки пользователем (Рис 1.2). Функция имеет следующий синтаксис:

**MsgBox(Сообщение, Кнопки, Заголовок),**

где

*Сообщение* – текст, максимальная длина 1024 символа. Перенос текста – клавиша Enter;

*Заголовок* – выражение, которое отображается в верхней строке окна. Если этот аргумент опущен, в строке будет имя приложения, например, MS Exce

*Кнопки* – выражение, показывающее, какие кнопки следует поместить в диалоговом окне.

Например:

**Ответ=MsgBox(“Обнаружена  
ошибка.Продолжить?”,VbYesNo)**

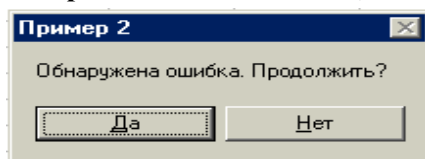


Рис. 1.2 Диалоговое окно функции MsgBox

Пользователь может назначать кнопки в зависимости от поставленной задачи, согласно таблице 1.1.

Таблица 1.1

Кнопки диалогового окна

Выражение	Кнопки	Значение
VbOkOnly или не указано	Ок	1
VbYesNo	Да, Нет	6, 7
VbCancel	Прервать,	3
	Повтор,	4
	Пропустить	5
VbYesNoCancel	Да, Нет, Отмена	6, 7, 2
VbOkCancel	Ок, Отмена	1, 2

**Пример 1.**

Написать программу для вычисления суммы двух целых чисел. Ввод и вывод данных организовать в диалоговом окне.

Порядок выполнения:

1. Запустить приложение MS Excel.
2. Открыть вкладку *Разработчик* и выбрать *Visual*

**Basic.**

3. В открывшемся окне выполнить команду *Insert-Module*.

4. Далее выполнить команду *Insert-Procedure*.

5. В открывшемся окне (Рис 1.3) ввести имя процедуры (длина не превышает 255 символов, начинается с буквы латинского или русского алфавита, не должно совпадать со стандартными именами VBA, не должно включать точек, пробелов, знаков ! # \$ % @ & + / - \*;

6. В открывшемся окне проекта ввести программный код (Рис 1.4).

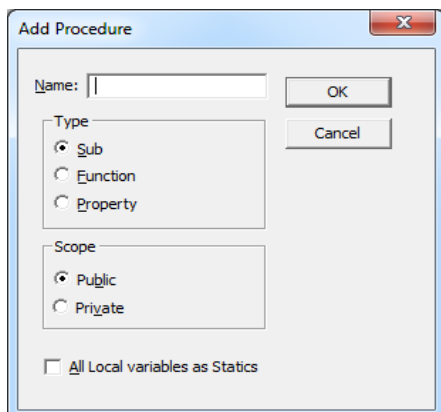


Рис. 1.3 Окно создания процедуры

```
Public Sub lab1()  
'Блок описания переменных  
Dim x As Integer  
Dim y As Integer  
Dim z As Integer  
'Блок ввода данных  
x = InputBox("Введите x")  
y = InputBox("Введите y")  
'Блок вычислений  
z = x + y  
'Блок вывода данных  
MsgBox ("z=" + Str(z))  
End Sub
```

Рис. 1.4 Программный код для вычисления суммы

В данном случае при выводе результата используется функция *Str()* которая позволяет перевести числовое значение в строковое и вставляет пробел впереди для положительных чисел.

7. Запуск программы осуществляется клавишей F5. Далее следуя диалогу необходимо ввести исходные данные.

8. Для сохранения созданного файла необходимо перейти в окно Excel и выполнить команду *Сохранить как* в меню



Файл. При сохранении выбрать тип файла книга Excel с поддержкой макросов.

## 1.2 Ввод и вывод данных в ячейки электронной таблицы MS Excel

### 1) Приемы ввода и вывода данных

Для того чтобы ввести в программу переменные, значение которых находится в ячейках электронной таблицы Excel необходимо обратиться к соответствующим листам и ячейкам следующим образом.

**A=Range("A4").Value** - переменной A присваивается значение, находящиеся в ячейке с адресом A4 текущего активного листа Excel. Функция Value в данном случае возвращает значение находящиеся в указанной ячейке.

**A=Worksheets(1).Range("A4").Value** - переменной A присваивается значение, находящиеся в ячейке с адресом A4 электронной таблицы листа с порядковым номером 1.

**A=Worksheets("Лист1").Range("A4").Value** - переменной A присваивается значение, находящиеся в ячейке с адресом A4 электронной таблицы листа с именем Лист1.

**A= Worksheets("Лист 1").Cells(1,2).Value** – переменной A присваивается значение, находящиеся в ячейке, находящейся на пересечении 1 строки и 2 столбца электронной таблицы листа с именем Лист1.

Аналогичном образом осуществляется вывод полученных результатов в ячейки электронной таблицы Excel.

**Range("A4").Value=A** – в ячейку с адресом A4 текущего активного листа Excel вводится значение переменной A.

**Worksheets(1).Range("A4").Value =A** в ячейку с адресом A4 электронной таблицы листа с порядковым номером 1 вводится значение переменной A.

**Worksheets("Лист1").Range("A4").Value=A** в ячейку с адресом A4 электронной таблицы листа с именем *Лист1* вводится значение переменной A.

**Worksheets("Лист1").Cells(1,2).Value=A** в ячейку, находящуюся на пересечении 1 строки и 2 столбца электронной таблицы листа с именем *Лист1* вводится значение переменной A.

## 2) Описание листа Excel как объекта

В процессе написания программного кода возможно описать лист электронной таблицы в качестве объекта. Это выполняется следующим образом.

**Dim Данные, Результат As Object** – объявляется тип переменных *Данные* и *Результат* как *Object*.

**Set Данные = Worksheets("Данные")** –назначается объект **Данные** как лист с именем *Данные*.

**Set Результат = Worksheets("Результат")** - назначается объект **Результат** как лист с именем *Результат*.

### Пример 2.

Написать программу для вычисления суммы двух целых чисел. Ввод и вывод данных организовать с помощью ячеек электронной таблицы на листах с именами *Данные* и *Результат*.

```

Public Sub lab2()
'Блок описания переменных
Dim x, y, z As Single
Dim Данные, Результат As Object
'Блок описания объектов
Set Данные = Worksheets("Лист1")
Set Результат = Worksheets("Лист2")
'Блок ввода данных
x = Данные.Range("A1").Value
y = Данные.Range("B1").Value
'Блок вычислений
z = x + y
'Блок вывода данных
Результат.Range("A1").Value = z
End Sub

```

Рис. 1.5 Программный код для вычисления суммы

### **Задание на самостоятельную работу**

В таблице Excel следует разместить сведения о результатах сдачи экзаменов студентами одной группы. Результаты заносятся в предварительно созданную таблицу (Рис 1.6).

	A	B	C	D
1	ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ			
2	Специальность		080502	
3	Фамилия И.О.	Номер зачетки	Дисциплина	Оценка
4				
5				

Рис. 1.6 Исходная таблица для записи данных

Данные должны передаваться в таблицу из программы.

В программу из таблицы нужно записать номер специальности, по которой обучаются студенты. Текст программы представлен на рис 1.7.

```
Public Sub Ведомость ()
' Описание переменных
Dim фамилия, Дисциплина As String
Dim НомерЗачетки, Оценка As Integer
Dim Специальность As Single
' Запись номера специальности из Excel в VBA
Специальность = Range("C2").Value
' Запись в программу сведений о студенте
фамилия = "Петров"
НомерЗачетки = 700111
Дисциплина = "физика"
Оценка = 4
' Передача данных из VBA в Excel
Range("A4").Value = фамилия
Range("B4").Value = НомерЗачетки
Range("C4").Value = Дисциплина
Range("D4").Value = Оценка
End Sub
```

Рис. 1.7 Программный код для заполнения таблицы

Изменить программу следующим образом. Организовать ввод фамилии студента через окно **InputBox**, записать эту фамилию в ячейку A4. После этого вывести через окно **MsgBox** номер специальности, прочитанный из ячейки C2. В окне должен быть заголовок «Номер специальности» и одна кнопка ОК.

## Индивидуальные задания

1. Вычислить площадь кольца. Наружный и внутренний радиусы вводить в диалоге. Результат записать на лист электронной таблицы с именем *Площадь* в ячейку A7.
2. Одна страница текста занимает объем 50 килобайт. Один символ – 2 байта. Вычислить количество символов с пробелами на одной странице. Объем вводить в диалоге, а количество символов записать на лист электронной таблицы с именем *Символ* в ячейку C10.
3. Вычислить площадь трапеции. Основания и высоту прочитать с листа электронной таблицы с именем *Трапеция* из ячеек B3, B4 и B5, а результат вывести в диалоге.
4. Рукопись содержит 100 страниц. На каждой странице 50 строк, в каждой строке 80 символов с пробелами. Один символ – 16 бит. Вычислить объем памяти рукописи в килобайтах. Исходные данные прочитать с листа электронной таблицы с именем *Рукопись* из ячеек A1:A3, а объем вывести в диалоговом окне.
5. Вычислить скорость велосипедиста  $v$ , если за время  $t$  он проехал путь  $S$ . Скорость и время прочитать с листа электронной таблицы из ячеек B2 и B3, а результат записать в ячейку B4.
6. Одна страница текста содержит 22800 символов с пробелами. Один символ – 2 байта. Вычислить объем в килобайтах. Количество символов прочитать с листа электронной таблицы с именем *Страница*, а объем вывести в диалоге.
7. Вычислить площадь поверхности цилиндра. Радиус и высоту вводить в диалоге, а площадь записать на лист электронной таблицы с именем *Цилиндр* в ячейку C5.
8. В окружность радиусом  $R1$  вписана окружность радиусом  $R2$ . Вычислить площадь кольца. Радиусы вводит в диалоге, а площадь записать на лист электронной таблицы с именем *Кольцо* в ячейку A4.

9. Вычислить, за какое время  $t$  автомобиль проедет путь  $S$ , двигаясь со скоростью  $v$ . Путь и скорость вводить в диалоге. Время вывести в диалоге.

10. Рукопись содержит 50 страниц. На каждой странице 50 строк, в каждой строке 80 символов с пробелами. Один символ - 18 бит. Вычислить объем памяти рукописи в килобайтах. Исходные данные прочитать с листа электронной таблицы с именем *Рукопись* из ячеек B1:B3, а объем вывести в диалоговом окне.

11. Вычислить скорость велосипедиста  $v$ , если за время  $t$  он проехал путь  $S$ . Скорость и время прочитать с листа электронной таблицы из ячеек A2 и A3, а результат записать в ячейку A4.

12. Вычислить площадь кольца. Наружный и внутренний радиусы вводить в диалоге. Результат записать на лист электронной таблицы с именем *Площадь* в ячейку B7.

13. Вычислить площадь трапеции. Основания и высоту прочитать с листа электронной таблицы с именем *Трапеция* из ячеек C3, C4 и C5, а результат вывести в диалоге.

14. Вычислить площадь поверхности цилиндра. Радиус и высоту вводить в диалоге, а площадь записать на лист электронной таблицы с именем *Цилиндр* в ячейку A5.

15. Одна страница текста занимает объем 50 килобайт. Один символ - 2 байта. Вычислить количество символов с пробелами на одной странице. Объем вводить в диалоге, а количество символов записать на лист электронной таблицы с именем *Символ* в ячейку A10.

## 2. Программирование разветвляющегося процесса

### 2.1 Операторы условного перехода

Для организации разветвлений в программе используется оператор перехода, который имеет структуру:

**If....**  
**Then....**  
**Else....**  
**End If**

После ключевого слова *If* записывают проверяемое условие. Если оно выполняется (ИСТИНА), осуществляется переход к операторам, расположенным после ключевого слова *Then*, в противном случае (ЛОЖЬ), осуществляется переход к операторам после ключевого слова *Else*.

Если число условий больше двух, то можно использовать конструкцию:

**If....Then....**  
**ElseIf....Then....**  
**Else....**  
**End If**

После ключевого слова *If* записывается первое проверяемое условие. Если оно не выполняется, второе условие запишется после ключевого слова *ElseIf*.

#### Пример 3

Создать процедуру вычисляющую значение *У*. Значение *Х* ввести в диалоговом окне, значение *У* записать на лист ЭТ в ячейку А7.

$$y = \begin{cases} 0.7x^2, & \text{если } x < 0 \\ 5 + x, & \text{если } x \geq 0 \end{cases}$$

```

Public Sub ДвеВетви()
X = InputBox("Введи X")
If X < 0 Then
Y = 0.7 * X ^ 2
Else
Y = 5 + X
End If
MsgBox(X)
Range("A7") = Y
End Sub

```

Рис.2.1 Программный код «Две ветви»

Для проверки работы программы необходимо дважды запустить её (F5). Ввести значение  $X < 0$ , перейти на вкладку ЭТ и получить результат, затем вернуться в окно кода и ввести  $X \geq 0$  – получить новый результат.

#### Пример 4.

Создать процедуру вычисляющую значение  $Y$ . Значение  $X$  прочесть с листа ЭТ из ячейки С2, а значение  $Y$  записать в ячейку С3 того же листа.

$$y = \begin{cases} x^3, & \text{если } x \leq -5 \\ \sin x, & \text{если } -5 < x \leq 0 \\ \sqrt{x}, & \text{если } x > 0 \end{cases}$$

```

Public Sub ТриВетви()
x = Range("C2")
If x <= -5 Then
Y = x ^ 3
ElseIf x <= 0 Then
Y = Sin(x)
Else
Y = Sqr(x)
End If
Range("C3") = Y
End Sub

```



Рис 2.2 Программный код «Три ветви»

Необходимо проверить работу программы по трём ветвям. Ввести значение X в соответствии с заданием и получить три результата.

## 2.2. Оператор безусловного перехода

Оператор безусловного перехода используется для передачи управления на некоторую заранее заданную строку программы.

Синтаксис оператора:

**GoTo Метка,**

где

*GoTo* – ключевое слово Visual Basic;

*Метка* – последовательность символов, заканчивающаяся двоеточием.

Метка может начинаться как с буквы, так и с цифры. Она может содержать символы латинского и русского алфавитов, цифры и знак подчеркивания. В метке нельзя использовать скобки, пробелы, знаки пунктуации и арифметических операций. Метка всегда ставится на отдельной строке перед каким-либо оператором программного кода. Помеченный оператор будет выполняться сразу после оператора GoTo. Такой способ передачи управления называется безусловным переходом, так как он выполняется без проверки каких-либо условий. Передавать управление таким способом можно как вперед, так и назад по тексту программы.

### Пример 5.

Вычислить

$$y = \begin{cases} \sin x, & \text{если } x > 3 \\ 4x, & \text{если } x \leq 3 \end{cases}$$

Значение X вводится с помощью окна InputBox, а результат следует поместить в ячейку C2 электронной таблицы.

После вывода результата в ячейку C2 в окне MsgBox вывести запрос «Продолжить вычисления?» и две кнопки: Yes и No. При нажатии на кнопку Yes повторить все действия программы (используйте оператор GOTO), в противном случае закончить выполнение программы.

```
Public Sub lab2()  
Dim x, y As Single  
M1: x = InputBox("Введите x")  
If x > 8 Then  
y = Cos(x)  
Else  
y = 2 * x  
End If  
Range("C2").Value = y  
R = MsgBox("Продолжить вычисления ", vbYesNo)  
If R = 6 Then GoTo M1  
Else  
End If  
End Sub
```

Рис. 2.2. Пример программы

## 2.3 Операторы выбора

В одном условном операторе можно использовать несколько проверок дополнительных условий. Кроме того, для проверки нескольких условий удобно использовать оператор выбора Select Case:

```
Select Case ИмяПеременной
Case Условие1 Оператор1
Case Условие2 Оператор2
.....
Case УсловиеN ОператорN
Case Else Оператор при невыполнении всех
условий
End Select
```

**Пример 6.** Создать процедуру вычисления значения У. переменную X прочитать с листа Первый из ячейки В7, а значение У вывести в диалоговом окне. При этом переменная X может

принимать только целые значения.

$$y = \begin{cases} x^2, & \text{если } x < -10 \\ 3 + x, & \text{если } -10 \leq x \leq -5 \\ \sin x, & \text{если } -5 < x \leq -2 \\ e^x, & \text{если } -2 < x \leq 0 \\ 8x, & \text{если } x > 0 \end{cases}$$

```

Public Sub Выбор()
Dim Первый As Object
Set Первый = Worksheets("Первый")
X = Первый.Range("B7")
Select Case X
Case Is < -10
Z = X ^ 2
Case -10 To -5
Z = 3 + X
Case -4 To -2
Z = Sin(X)
Case -1 To 0
Z = Exp(X)
Case Else
Z = 8 * X
End Select
MsgBox ("Значение Z = " & Z)
End Sub

```

Рис 2.4 Процедура «Выбор»

### Индивидуальные задания

1. Составить программу вычисления

$$y = \begin{cases} x-5, & \text{если } x < 0 \\ x^2, & \text{если } 0 \leq x < 7, \\ \sqrt{x}, & \text{если } x \geq 27 \end{cases}$$

Значение x прочитать с листа ЭТ «Первый» из ячейки A1.

Результат поместить на лист «Второй» в ячейку A2.

2. Составить программу вычисления

$$y = \begin{cases} x+10, & \text{если } x < 0 \\ x^2, & \text{если } 10 \leq x < 27 \\ \sqrt{x}, & \text{если } x \geq 27 \end{cases}$$

Значение x ввести в диалоговом окне. Результат поместить на лист «Результат» в ячейку A2.

3. Составить программу вычисления

$$y = \begin{cases} x^4, & \text{если } x \leq -20 \\ x + 10, & \text{если } -20 \leq x \leq -10 \\ \sin x & \text{если } -10 < x \leq -10 \\ x^3 + 3, & \text{если } 0 < x \leq 5 \\ 9x, & \text{если } x > 5 \end{cases}$$

Значение  $x$  прочесть с листа ЭТ «Первый» из ячейки А1. Вывести в диалоговом окне.

4. Составить программу вычисления

$$y = \begin{cases} x^2, & \text{если } x \leq 0 \\ x - 3, & \text{если } 0 \leq x \leq 4 \\ \ln x, & \text{если } 4 < x \leq 8 \\ e^x + 5x, & \text{если } 8 < x \leq 15 \\ \cos x, & \text{если } x > 15 \end{cases}$$

Значение  $x$  ввести в диалоговом окне. Результат поместить на лист «Результат» в ячейку А2.

5. Составить программу начисления премии сотрудникам, превышающим норму: на 15% - премия 2000 руб, на 25% -5000 руб, на 35% -8000 руб, на 45% -10000 руб, если больше 45%- премия 0,5 от оклада. Исходные данные (%) прочесть с листа «Процент» из ячейки В3, а результат записать на лист премия в ячейку В3.

6. Составить программу вычисления скидок на товары: стоимостью более 36000- скидка 7%; стоимостью от 26000 до 35000 –скидка 10%; стоимостью от 16000 до 25000 –скидка 12%; стоимостью от 10000 до 15000 –скидка 13%.

Исходные данные стоимость товара прочесть с листа ЭТ из ячейки В3, а результат вывести в диалоговом окне.

7. Составить программу выпуска количества прохладительных напитков в зависимости от времени года. Зима -2,5 млн. л; Весна - 3,5 млн. л; Лето-5 млн л; Осень-3 млн.л.

Исходные данные( время года прочесть из ячейки А1 , а

количество напитков вывести в диалоговом окне.

8. Составить программу начисления стипендии студенту в зависимости от среднего балла. Если средний балл  $>1.5$  то стипендия 2000 руб; если средний балл находится в пределах  $[3.5 ; 4.5]$  то 1500; в остальных случаях стипендии нет. Исходные данные прочитать из ячейки В3, а стипендию вывести в диалоговом окне.

9. Составить программу вычисления

$$y = \begin{cases} x^3, & \text{если } x \leq 1 \\ x + 8, & \text{если } 1 < x \leq 5 \\ \cos x, & \text{если } 5 < x \leq 10 \\ x^2 - 2x, & \text{если } 10 < x \leq 12 \\ 7x, & \text{если } x > 12 \end{cases}$$

Значение  $x$  ввести в диалоговом окне. Результат поместить на лист «Результат» в ячейку А1.

10. Составить программу вычисления

$$y = \begin{cases} e^{-x}, & \text{если } x \leq -10 \\ |x|, & \text{если } -10 < x \leq -5 \\ \sin x, & \text{если } -5 < x \leq 2 \\ \ln x, & \text{если } 2 < x \leq 6 \\ x + 5, & \text{если } x > 6 \end{cases}$$

Значение  $x$  ввести в диалоговом окне. Результат поместить на лист «Результат» в ячейку А2.

11. Составить программу начисления премии сотрудникам, превышающим норму: на 20% - премия 5000 руб, на 40% -10000 руб, на 60% -15000 руб, на 70% -25000 руб, если больше 70%-премия 100% от оклада. Исходные данные (%) прочитать с листа «Процент» из ячейки В3, а результат записать на лист премия в ячейку В3.

12. Составить программу вычисления скидок на товары: стоимостью более 20000- скидка 20%; стоимостью от 20000 до 15000 –скидка 15%; стоимостью от 15000 до 10000 –скидка 10%; стоимостью от 10000 до 5000 –скидка 5%.

Исходные данные стоимость товара прочитать с листа ЭТ из ячейки В3, а результат вывести в диалоговом окне.

13. Составить программу вычисления

$$y = \begin{cases} x-6, & \text{если } x < 0 \\ x^2, & \text{если } 0 \leq x < 17, \\ \sqrt{x^3}, & \text{если } x \geq 17 \end{cases}$$

Значение  $x$  прочитать с листа ЭТ «Первый» из ячейки А1. Результат поместить на лист «Второй» в ячейку А2.

14. Составить программу вычисления

$$y = \begin{cases} x+2, & \text{если } x < 0 \\ x^4, & \text{если } 5 \leq x < 15 \\ 2\sqrt{x}, & \text{если } x \geq 15 \end{cases}$$

Значение  $x$  прочитать с листа ЭТ «Первый» из ячейки А1. Результат поместить на лист «Второй» в ячейку А2.

15. Составить программу вычисления

$$y = \begin{cases} x-4, & \text{если } x < 2 \\ x^3, & \text{если } 2 \leq x < 10 \\ \sqrt{5x}, & \text{если } x \geq 10 \end{cases}$$

Значение  $x$  ввести в диалоговом окне. Результат поместить на лист «Результат» в ячейку А2.

### 3. Программирование циклического процесса

#### 3.1. Операторы цикла с параметром – счетчиком шагов

Для организации циклического процесса со счетчиком шагов используются операторы цикла, имеющие следующую структуру:

**For** Имя Переменной=Начало Цикла **To** Конец Цикла  
**Step** Шаг Цикла

Операторы Тела Цикла

**Next**

где

- **For** – ключевое слово Visual Basic, обозначающее начало цикла;
- Имя переменной – определенная пользователем переменная счетчика цикла;
- Начало Цикла – начальное значение счетчика;
- **To** – ключевое слово Visual Basic;
- Конец Цикла – значение счетчика, после которого цикл завершается;
- **Step** – ключевое слово, используемое при указании шага цикла, необязательный аргумент;
- Шаг Цикла – значение шага цикла, т.е. значение, на которое увеличивается значение счетчика на каждом шаге. Это число может быть отрицательным;
- **Next** оператор, обозначающий конец цикла «For». Указывать в нем переменную счетчика не обязательно, хотя рекомендуется;
- **Exit For** – оператор принудительного завершения цикла.

Работа цикла происходит следующим образом:

- переменной цикла присваивается начальное значение ;
- выполняются **Операторы Тела Цикла**;
- оператор **Next** возвращает управление оператору **For**;
- оператор **For** увеличивает значение переменной цикла на шаг и проверяет условие окончания цикла;
- если в теле цикла встречается оператор **Exit For**, то происходит принудительный выход из текущего цикла (цикл



прерывается и управление передается оператору следующим за оператором *Next*)

При использовании вложенных циклов закрывается вначале внутренний, а затем внешний цикл. Если шаг цикла равен 1, то оператор *Step* опускают.

**Пример 7** . Написать процедуру для нахождения суммы ряда:

$$\sum_{n=1}^{100} \frac{1}{n^2}$$

```

Public Sub Сумма100()
Dim n As Integer
'Переменная n - целое число
Dim S As Single
'Переменная S - вещественное число
S = 0
'Обнуление начальной суммы S
'Выполнение цикла
For n = 1 To 100
S = S + 1 / n ^ 2
Next
'Вывод результата в диалоговом окне
MsgBox ("Сумма 100 = " & S)
End Sub

```

Рис. 3.1 Процедура суммирования

### 3.2 Операторы цикла с условием Do ... Loop

Циклом с условием является цикл «**Do...Loop**». Такие циклы также называют *циклами с неизвестным числом повторений* или *итерационными*. Он продолжает свою работу в зависимости от состояния условия. Главной особенностью таких циклов является наличие условия, т.е. любого логического выражения, функции или переменной, принимающих значения True (ИСТИНА) или False (ЛОЖЬ). Двумя основными циклами с условиями являются:

цикл «**Do...While**», который выполняется *до тех пор, пока условие сохраняет значение True*, и

цикл «**Do...Until**», работающий, *пока условие не станет равным True*.

Для вышеописанных циклов различают *циклы с пост- и предусловием*. Различаются они тем, где именно в цикле записывается условие, при этом принцип работы циклов не меняется.

Таблица 3.1

Синтаксис итерационных циклов

	Do...While	Do...Until
С предусловием	<b>Do While</b> Условие <b>Операторы</b> <b>Loop</b>	<b>Do Until</b> Условие <b>Операторы</b> <b>Loop</b>
С постусловием	<b>Do</b> <b>Операторы</b> <b>Loop While</b> Условие	<b>Do</b> <b>Операторы</b> <b>Loop Until</b> Условие

Единственным отличием циклов с пред- и постусловием является то, что в первом случае тело цикла может не выполнить ни одного раза, а во втором случае тело цикла всегда выполнится хотя бы один раз.

**Пример 8 .** Найти сумму ряда

$$\sum_{n=1}^{100} \frac{1}{n^2}$$

Суммирование продолжать до тех пор, пока сумма не достигнет значения 1,6.

```
Public Sub Сумма100()
Dim n As Integer
'Переменная n - целое число
Dim S As Single
'Переменная S - вещественное число
S = 0
'Обнуление начальной суммы S
'Выполнение цикла
For n = 1 To 100
S = S + 1 / n ^ 2
Next
'Вывод результата в диалоговом окне
MsgBox ("Сумма 100 = " & S)
End Sub
```

Рис. 3.2 Процедура суммирования с условием

На рисунке 3.3 представлен код программы для решения Примера 8 с помощью оператора Do Until...Loop.

```
Public Sub СуммаРавнаЧислуUntil()  
Dim n As Integer  
Dim S As Single  
n=1  
S=0  
'Организация цикла – суммирование  
продолжать, пока сумма не достигнет  
значения 1,6  
Do Until S >= 1.6  
S=S+1/(n^2)  
n=n+1  
Loop  
MsgBox("Сумма равна " & S )  
End Sub
```

Рис. 3.3 Процедура суммирования с условием

Если возникает необходимость досрочного прерывания цикла **Do...Loop**, то необходимо использовать в необходимом месте тела цикла **Exit Do**. Как только компьютер встречает этот оператор, то цикл прерывается и управление передается оператору следующим за оператором **Loop**.

### 3.3 Оператор цикла с условием **While... Wend**

Оператор повторяет набор инструкций, пока условие выполняется.

**While** Условие=**ИСТИНА**

Операторы тела цикла

**Wend**

На рисунке 3.4 представлен код программы для решения Примера 8 с помощью данного оператора.

```
Public Sub СуммаРавнаЧислу()  
Dim n As Integer  
Dim S As Single  
n=1  
S=0
```

*'Организация цикла – суммирование продолжать,*

пока сумма не достигнет значения 1.6

```
While S < 1.6
    S=S+1/(n^2)
    n=n+1
Wend
MsgBox("Сумма равна " & S)
End Sub
```

Рис. 3.4 Процедура суммирования с условием

### 3.4 Табулирование функций

Табулирование функции - это вычисление значений функции при изменении аргумента от некоторого начального значения до некоторого конечного значения с определенным шагом.

**Пример 9.** Разработать приложение для табулирования функции  $y = x^2 e^{-x} \sin x$  в интервале  $[-3, 2]$  с шагом 0.5

```
Public Sub пример_2()
Dim s As String
s = " x      y"
For x = -3 To 2 Step 0.5
y = x ^ 2 * Exp(-x) * Sin(x)
s = s + vbNewLine + Format(x, "0.0") + "    " + Format(y, "#.00")
Next x
MsgBox (s)
End Sub
```

Рис. 3.5. Табулирование функции

Результат работы данной процедуры выводится в диалоговое окно (Рис 3.6).

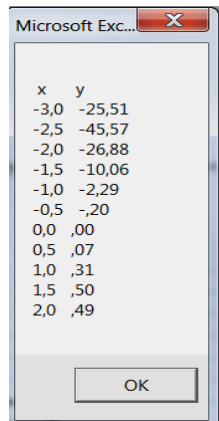


Рис.3.6 Результат работы процедуры

**Пример 9.** Создать программу для вычисления

$$S = \sum_{i=1}^n f(x)$$

где  $f(x)=ax+4b$ .

Значение  $a$  и  $b$ , начальное и конечное значение  $x$ , количество точек пользователь задаёт самостоятельно.

```

Public Sub summ()
' Раздел описания переменных
Dim x, x0, y, xn, s, h As Single
Dim i, n As Integer
' Очистка ячеек
Worksheets("Лист1").Cells.Clear
' Вывод надписей x и y в ячейках
Worksheets("Лист1").Range("A1").Value = "x"
Worksheets("Лист1").Range("B1").Value = "y"
' Ввод данных

a = InputBox("Введите a")
b = InputBox("Введите b")
n = InputBox("Введите количество точек")
x0 = InputBox("Введите начальное значение X")
xn = InputBox("Введите конечное значение X")
h = (xn - x0) / n 'Вычисляем шаг
i = 2 'Номер строки с которой начинается заполнение ячеек
s = 0
' Цикл для расчета значения функции и суммы
For x = x0 To xn Step h
y = a * x + 4 * b
s = s + y
Worksheets(1).Cells(i, 1).Value = x 'Вывод значений x и y на лист Excel
Worksheets(1).Cells(i, 2).Value = y
i = i + 1
Next x
' Вывод суммы
Range("C1").Value = "Сумма"
Range("C2").Value = s
End Sub

```

Рис. 3.7. Пример программы

### Индивидуальное задание

#### Задание 1.

Создать программу для вычисления  $s = \sum_{i=1}^n f(x)$  и

Где  $f(x)=ax+4b$

В процессе работы необходимо выполнить следующее:

1. Вывести диалоговые окна для задания начального, конечного значения  $x$  и шага  $h$ . Шаг рассчитать в зависимости от заданного количества точек.
2. На листе Excel создать таблицу значений  $x$  и  $y$ , которая обновлялась бы каждый раз при запуске программы.
3. Найти  $y$  в зависимости от значений  $x$ .
4. Вывести значения  $x$ ,  $y$  и суммы на лист Excel
5. Варианты задания приведены в таблице 3.1

Таблица 3.1

№	$n$	$f(x)$	№	$n$	$f(x)$
1	10	$ax + b$	9	14	$ax - 6b^2$
2	12	$ax^2 + 3b$	10	12	$a\sqrt{x - 7b}$
3	15	$ax - 4b$	11	18	$ax - 5b^4$
4	8	$a\sqrt{x} - 5b$	12	16	$ax + 6b$
5	15	$x^2 - ab^2$	13	14	$2ax - 4b$
6	12	$ax^3 + 4b$	14	12	$4ax^2 - 2b$
7	11	$a\sqrt{x} - 2b$	15	15	$\sqrt{x} - 2ab$
8	9	$\sqrt[3]{x} - 2ab$	16	11	$ax^4 + 4b$

## Задание 2

Создать программу для сравнения  $\sum_{i=1}^n y(x)$  и  $\sum_{i=1}^n y_1(x)$

В процессе работы необходимо выполнить следующее:

1. Вывести диалоговые окна для задания значений начального, конечного значения  $x$  и шага  $h$
2. На листе Excel создать таблицу значений  $x$ ,  $y$ ,  $y_1$  которая бы обновлялась каждый раз при запуске программы.
3. Найти  $y$  и  $y_1$  в зависимости от значений  $x$ .
4. Вычислить суммы всех значений  $y$  и  $y_1$ , поместить их в соответствующие ячейки.
5. Сравнить значения полученных сумм и вывести сообщение в диалоговом окне, какая сумма больше.
6. Варианты задания приведены в таблице 3.2.



Таблица 3.2

№	$y(x)$	$y_1(x)$	№	$y(x)$	$y_1(x)$
1	$x^2$	$e^x$	9	$\sqrt{x} - 4x$	$x^3$
2	$x^4$	$\sqrt{x} - 2x$	10	$\sqrt{x} - 2x$	$6x^2$
3	$x^3$	$e^x$	11	$8x^2$	$e^x$
4	$x^3$	$6x^2$	12	$x^3 - 2x^2$	$2x^3$
5	$x^2 - 2x$	$x^3 - 2x^2$	13	$2x^3 - x^2$	$4x^3$
6	$x^3 + 2x$	$e^x$	14	$x^3 + 2x^2$	$4e^x$
7	$x^3 - 2x^2$	$x^4$	15	$\sqrt{x} - 4x$	$3x^4$
8	$e^{2x}$	$2x^3$	16	$x^3 + x^2$	$x^2$

### 3.5 Работа с массивами

Массив это упорядоченный набор данных разного типа (числового, текстового и др.), который характеризуется:

- фиксированным набором элементов одного и того же типа;
- каждый элемент имеет уникальный набор значений индексов;
- количество индексов определяют мерность массива, например, два индекса - двумерный массив, три индекса - трехмерный массив, один индекс - одномерный массив или вектор;
- обращение к элементу массива выполняется по имени массива и значениям индексов для данного элемента

Массив с заданным изначально размером называется массивом *фиксированного размера или статическим*. Массив с переменным размером называется *динамическим*

Статический массив объявляется в начале программы следующим образом:

**Dim Имя Массива(Размерность)  
As Тип Элементов**

*Динамические массивы* в VBA объявляются в начале процедур без указания размерности:

**Dim B() As String**

Когда нужная размерность массива становится известна, массив переопределяется с помощью оператора

**ReDim B(25)**

**Пример 10.** Вычислить среднее значение элементов массива M(5)

```
Public Sub Среднее()  
Dim M(5) As Integer  
Summa = 0  
'Чтение и суммирование  
'всех элементов массива  
'в цикле из 5  
For i = 1 To 5  
M(i) = Cells(i, 1)  
Summa = Summa + M(i)  
Next  
'Вычисление среднего  
Srednee = Summa / 5  
'Вывод в ячейку B1  
Range("B1") = Srednee  
End Sub
```

Рис. 3.8. Пример программы

**Пример 11.** Найти произведение элементов массива A(10).

```
Public Sub Pr()  
Dim A(10) As Integer  
p = 1  
For i = 1 To 10  
A(i) = InputBox("Введите элемент массива")  
p = p * A(i)  
Next i  
MsgBox ("Произведение =" & p)  
End Sub
```

Рис. 3.9. Пример программы

**Пример 12.** Вычислить сумму элементов массивов

```

Public Sub Pr()
Dim A(7), B(7), C(7) As Integer
For i = 1 To 7
A(i) = Worksheets("Данные").Cells(3, i)
B(i) = Worksheets("Данные").Cells(4, i)
C(i) = A(i) + B(i)
Worksheets("Результат").Cells(i, 2) = C(i)
Next i
End Sub

```

Рис. 3.10. Пример программы

**Пример 13.** Найти номер первого отрицательного элемента массива

```

Public Sub Pr()
Dim A(10) As Single
For i = 1 To 10
A(i) = Cells(i + 1, 3)
Next i
N = 1
While A(N) >= 0
N = N + 1
Wend
MsgBox ("Номер отрицательного" & N)
End Sub

```

Рис. 3.11. Пример программы

**Пример 14.** Найти сумму положительных элементов в массиве до тех пор, пока не встретиться отрицательный элемент

```

Public Sub Pr()
Dim A(10) As Single
For i = 1 To 10
A(i) = Cells(i, 1)
Next i
S = 0
i = 1
Do Until A(i) < 0
If A(i) > 0 Then
S = S + A(i)
Else
S = S
End If
i = i + 1
Loop
MsgBox ("Сумма равна" & S)
End Sub

```

Рис. 3.12. Пример программы

**Пример 15** . Найти сумму элементов массива K(270). Суммирование продолжать до тех пор, пока сумма не достигнет значения 3000.

```

Public Sub p()
Dim K(270), i, s As Variant
s = 0
i = 1
While s <= 3000 And i <= 270
K(i) = Cells(i, 1)
s = s + K(i)
i = i + 1
Wend
MsgBox (" Сумма равна " & s)
MsgBox (" Суммируется " & i - 1 & " элементов")
End Sub

```

Рис. 3.13. Пример программы

### 3.6. Вложенные циклы

В некоторых случаях важно повторить подзадачу несколько раз внутри более общей задачи. Один из способов написания такой программы – включить в цикл набор инструкций (*внутренний цикл*), которые повторяются внутри другого цикла (*внешний цикл*). Такая структура, состоящая из цикла в цикле, называется *вложенными циклами*. Любой цикл может быть вложен в любой другой

независимо от их видов. Например, можно вложить цикл **For** в такой же цикл **For** или в любой цикл **Do...Loop** и наоборот:

```
Do While condition  
For CounterVar = StartNum To EndNum  
...  
Next CounterVar  
Loop
```

Особенно часто с вложенными циклами приходится сталкиваться при работе с многомерными массивами.

**Пример 16.** Вычислить количество отрицательных и положительных элементов в массиве.

```
Public Sub Pr()  
Dim A(6, 7) As Variant  
Плюс = 0  
Минус = 0  
For i = 1 To 6  
For j = 1 To 7  
A(i, j) = Cells(i, j)  
If A(i, j) >= 0 Then  
Плюс = Плюс + 1  
Else  
Минус = Минус + 1  
End If  
Next j  
Next i  
MsgBox ("+" & Плюс & vbNewLine & "-" & Минус)  
End Sub
```

Рис. 3.14. Пример программы

**Пример 17.** Найти минимальный и максимальный элемент массива.

```

Public Sub МакМин()
Dim B(5, 4) As Integer
Dim Массив As Object
Set Массив = Worksheets("Массив")
Максимум = B(1, 1)
Минимум = B(1, 1)
For i = 1 To 5
For j = 1 To 4
B(i, j) = Массив.Cells(i + 3, j + 2)
If B(i, j) >= Максимум Then
Максимум = B(i, j)
ElseIf B(i, j) <= Минимум Then
Минимум = B(i, j)
End If
Next
Next
Массив.Range("A1") = Максимум
Массив.Range("A2") = Минимум
End Sub

```

Рис. 3.15. Пример программы

**Пример 18.** Найти сумму элементов каждой строки массива.

```

Public Sub prg()
Dim P(1 To 2, 1 To 5) As Integer, S(1 To 2) As Integer
Dim I As Integer, J As Integer
Dim Первый As Object
Set Первый = Worksheets("Первый")
For I = 1 To 2
For J = 1 To 5
P(I, J) = Первый.Cells(I, J)
Next
Next
For I = 1 To 2
S(I) = 0
For J = 1 To 5
S(I) = S(I) + P(I, J)
Next
Next
MsgBox ("Сумма" & I & "-й строки равна" & S(I))
Next
End Sub

```

Рис. 3.16. Пример программы

### Задание на самостоятельную работу.

Составить программу, заполняющую диапазон ячеек A1:D4 элементами матрицы по следующему правилу. На главной диагонали должны стоять единицы, во всех ячейках ниже главной диагонали – нули, выше главной диагонали – числа, каждое из которых является суммой номера столбца и номера строки.

#### Индивидуальное задание

1. Массив A(10) расположен на листе ЭТ *Первый* в ячейках B1:B10. Вычислить значения массива C(10), каждый элемент которого получается возведением в куб элементов массива A. Разместить массив C(10) в ячейки B10:K10 листа с именем *Второй*.

2. Вычислить сумму элементов массива B(15). Вычисления следует прекратить, когда сумма достигнет величины 250. Исходные данные взять из диапазона ячеек B1:B15 листа *Исходные*. Сумму поместить в ячейку A2 листа *Данные*.

3. Массив N(4,5) расположен в ячейках C2:G5 листа ЭТ *Задание*. Найти произведение элементов каждого столбца массива и записать в массив P(5). Вывести окно с сообщением «Произведение j-го столбца равно P(j)».

4. Вычислить сумму элементов массива C(10). Вычисления следует прекратить, когда сумма достигнет величины 100. Исходные данные взять из диапазона ячеек B1:B10 листа *Массив*. Сумму поместить в ячейку A15 листа *Сумма*.

5. Массив M(3,4) расположен в ячейках A1:D3 листа ЭТ *Начало*. Найти сумму элементов каждой строки массива и записать в массив S (5). Вывести окно с сообщением «Сумма i-го столбца равно S(i)».

6. Массив M(3,4) расположен в ячейках A1:D3 листа ЭТ *Максимум*. Найти максимальный элемент. Вывести результат в диалоговом окне.

7. Вычислить сумму элементов массива C(20). Вычисления следует прекратить, когда сумма достигнет величины - 350. Исходные данные взять из диапазона ячеек B1:B20 листа *Отрицательный*. Сумму поместить в ячейку A2 листа *Сумма*.

8. Вычислить количество отрицательных элементов массива  $C(20)$ . Исходные данные взять из диапазона ячеек  $B1:B20$  листа *Исходные*, результат вывести в диалоговом окне
9. Массив  $M(3,4)$  расположен в ячейках  $A1:D3$  листа ЭТ *Минимум*. Найти минимальный элемент. Вывести результат в диалоговом окне.
10. Массив  $B(10)$  расположен в ячейках  $A1:A10$  листа ЭТ *Начало*. Найти сумму отрицательных элементов. Вывести результат в диалоговом окне.
11. Массив  $M(5)$  расположен в ячейках  $A1:A5$  листа ЭТ *Среднее*. Вычислить среднее арифметическое элементов массива и записать результат в ячейку  $B1$  того же листа.
12. Массив  $A(8)$  расположен в ячейках  $C5:J5$  листа ЭТ *Произведение*. Вычислить произведение всех элементов массива. Вывести результат в диалоговом окне.
13. Массив  $B(5,4)$  расположен в ячейках  $C4:F8$  листа ЭТ *Массив*. Найти максимальный и минимальный элементы. Записать результаты в ячейки  $A1$  и  $A2$  того же листа.
14. Массив  $A(10)$  расположен в ячейках  $B14:K14$  листа ЭТ *Массив*. Найти сумму положительных элементов массива. Вывести результат в диалоговом окне.
15. Массивы  $A(7)$  и  $B(7)$  расположены в ячейках  $E3:K3$  и  $E6:K6$  листа ЭТ *Данные*. Получить массив  $C(97)$  путём сложения соответствующих элементов массивов  $A$  и  $B$ . Результат записать в ячейку  $D3:D9$  листа *Результат*.



### **Библиографический список.**

1. Уокенбах Д. MS Excel 2010: профессиональное программирование в VBA.- М.: Вильямс, 2012. -944 с.
2. Фризен, И. Г. Офисное программирование : учебное пособие. - Ростов-на-Дону: Феникс, 2010. - 241 с.
3. Векшина Н.В. Информатика. Прикладное программирование на языке VISUAL BASIC. Методические указания к лабораторным работам и самостоятельной работе для студентов всех направлений.- СПб: Национальный минерально-сырьевой университет «Горный», 2014.- 36 с.
4. Ильин А.Е., Кротова С.Ю., Чиргин А.В. ИНФОРМАТИКА. Программирование на VBA. Обработка и визуализация данных. Методические указания к лабораторным работам.- СПб, 2019. 45 с.
5. Кудрявцев Н.Г., Кудин Д.В., Беликова М.Ю. Программирование на VBA MS Excel. Учебное пособие. Горно-Алтайск. РИО Горно-Алтайского госуниверситета, 2015.-116 с.

## Содержание

Введение.....	3
1. Программирование линейного процесса.....	4
1.1 Функции ввода и вывода в диалоговые окна VBA.....	4
1.2 Ввод и вывод данных в ячейки электронной таблицы MS Excel .....	8
2. Программирование разветвляющегося процесса .	14
2.1 Операторы условного перехода .....	14
2.2. Оператор безусловного перехода.....	16
2.3 Операторы выбора.....	18
3. Программирование циклического процесса .....	23
3.1. Операторы цикла с параметром – счетчиком шагов .	23
3.2 Операторы цикла с условием Do ... Loop .....	25
3.4 Табулирование функций .....	28
3.5 Работа с массивами .....	32
3.6. Вложенные циклы .....	35
Библиографический список.....	40

## **ИНФОРМАТИКА**

### **ПРОГРАММИРОВАНИЕ НА VBA. ЛИНЕЙНЫЕ, РАЗВЕТВЛЯЮЩИЕСЯ И ЦИКЛИЧЕСКИЕ ПРОЦЕССЫ**

*Методические указания к курсовой работе  
для студентов бакалавриата направлений 13.03.02, 20.03.01, 27.03.01*

Сост.: *С.Ю. Кротова, А.Е. Ильин, А.В. Чиргин*

Печатается с оригинал-макета, подготовленного кафедрой  
информатики и компьютерных технологий

Ответственный за выпуск *С.Ю. Кротова*

Лицензия ИД № 06517 от 09.01.2002

Подписано к печати 06.02.2023. Формат 60×84/16.  
Усл. печ. л. 2,4. Усл.кр.-отг. 2,4. Уч.-изд.л. 2,1. Тираж 50 экз. Заказ 72.

Санкт-Петербургский горный университет  
РИЦ Санкт-Петербургского горного университета  
Адрес университета и РИЦ: 199106 Санкт-Петербург, 21-я линия, 2