

Тверской государственный технический университет

Е.Е. Фомина

**Программирование на языке VBA в MS Excel.
Примеры и задачи**

Тверь 2012

УДК 004.43(075.8)

ББК 22.18я7

К.т.н. Фомина, Е.Е. Программирование на языке VBA в MS Excel. Примеры и задачи [Электронный ресурс] / Е.Е. Фомина. Тверь, 2012. 46 с.

VBA представляет собой сочетание одного из самых простых языков программирования Basic со специальным механизмом, позволяющим программам, написанным на этом языке, обращаться к объектам основных приложений Microsoft Office – Excel, Word, Power Point, Access и др.

Методическое пособие содержит теоретический материал, охватывающий основные разделы по теме «Программирование на языке VBA», а также варианты заданий для самостоятельной работы.

Предназначено для студентов всех специальностей, изучающих курс «Информатика».

Обсуждено и рекомендовано на заседании кафедры ИПМ (протокол №8 от 29 декабря 2011 г.)

Рецензенты: кандидат физико-математических наук, доцент кафедры «Информатика и прикладная математика» ТГТУ Кислова И.Л.

К.т.н. Елена Евгеньевна Фомина

© Тверской государственный технический университет, 2012

Содержание

Характеристика языка VBA	4
<i>Встроенные типы данных</i>	<i>4</i>
<i>Переменные.....</i>	<i>5</i>
<i>Константы</i>	<i>6</i>
<i>Массивы</i>	<i>6</i>
<i>Ввод и вывод данных</i>	<i>8</i>
Ввод данных	8
Вывод данных.....	9
<i>Встроенные функции VBA</i>	<i>10</i>
Математические функции и арифметические операторы.....	11
Функции преобразования данных	11
<i>Запуск редактора VBA. Окно редактора VBA</i>	<i>12</i>
<i>Синтаксис процедуры</i>	<i>14</i>
Пример 1.....	14
Пример 2.....	16
<i>Операторы языка VBA.....</i>	<i>18</i>
Оператор комментария	18
Условный оператор	18
а) неполный оператор	18
Пример 3.....	19
б) полный оператор	21
Пример 4.....	22
Пример 5.....	24
Пример 6.....	27
<i>Операторы цикла</i>	<i>30</i>
а) Цикл со счетчиком (For ... Next)	30
Пример 7.....	31
б) Цикл с предусловием (Do While- Loop)	33
Пример 8.....	34
<i>Создание кнопок быстрого вызова</i>	<i>36</i>
Задания для лабораторных работ.....	39
<i>Лабораторная работа «Линейные вычислительные процессы»</i>	<i>39</i>
<i>Лабораторная работа «Разветвляющиеся вычислительные процессы»</i>	<i>40</i>
<i>Лабораторная работа «Циклические вычислительные процессы»</i>	<i>40</i>

Характеристика языка VBA

Встроенные типы данных

Данные, которые обрабатывает программа, могут быть разных типов, например, числовые, текстовые, дата и др. При работе с данными их значения необходимо хранить в памяти компьютера (ПК). Размер выделяемой памяти будет зависеть от типа информации.

VBA поддерживает следующие основные типы данных (табл. 1).

Таблица 1. Типы данных

Тип данных	Описание	Возможные значения	Требуемая память в байтах
<i>Boolean</i>	Данные логического типа	<i>true</i> (истина) <i>false</i> (ложь)	2
<i>Byte</i>	Натуральные числа	Диапазон значений: 0...255	1
<i>Integer</i>	Целые числа	Диапазон значений: -32768...+32767	2
<i>Single</i>	Действительные числа	Диапазон значений: $-3,4 \cdot 10^{38} \dots -1,4 \cdot 10^{-45}$ $1,4 \cdot 10^{-45} \dots 3,4 \cdot 10^{38}$	4
<i>String</i>	Текстовые данные	Строки длиной до 2^{16} символов	10
<i>Date</i>	Дата и время	Диапазон значений дат: 01.01.100 – 31.12.9999	8
<i>Variant</i>	Данные любого типа		≥ 16

Переменные

Переменная – это именованная область памяти ПК, предназначенная для хранения данных во время выполнения программы.

Имя переменной должно начинаться с буквы и представлять собой непрерывную последовательность букв, цифр и знаков подчеркивания.

Примеры имен переменных:

VAR1

R_1

example_4

x

y1

h_y

В качестве имени переменной нельзя использовать ключевые слова, такие как *function*, *single* и другие.

Для объявления переменной используется команда, которая имеет следующую конструкцию:

DIM <имя_переменной> **AS** <тип_переменной>

Например:

Dim x As Integer

При обращении к этой команде в памяти ПК будет выделена область с именем *x*, объемом 2 байта (т.к. тип переменной *Integer*, см. табл. 1), предназначенная для хранения данных целочисленного типа.

Dim st_1 As String

При обращении к этой команде в памяти ПК будет выделена область с именем *st_1*, объемом ≈10 байт (т.к. тип переменной *String*, см. табл. 1), предназначенная для хранения данных строкового типа.

Перед запуском программы необходимо описать все переменные, которые в ней используются!

Константы

Константа – это именованная область памяти ПК, содержащая значение, которое не изменяется во время выполнения программы.

Имя константы должно начинаться с буквы и представлять собой непрерывную последовательность букв, цифр и знаков подчеркивания.

Например:

g_x

const_1

pi

В качестве имени константы нельзя использовать ключевые слова, такие как *function*, *single* и другие.

Для объявления константы используется команда, которая имеет следующую конструкцию:

DIM <имя_константы> **AS** <тип_константы>=<значение>

Например:

Dim g As Single=9.8

При обращении к этой команде в памяти ПК будет выделена область с именем *g*, объемом 4 байта (т.к. тип переменной *Single*, см. табл. 1), предназначенная для хранения значения 9,8.

Перед запуском программы необходимо описать все константы, которые в ней используются!

Массивы

Массив – это именованная область памяти ПК, предназначенная для хранения совокупности данных одного типа.

Одномерный массив *A*, состоящий из 5 элементов.

Массив *A*

номера элементов	0	1	2	3	4

Двумерный массив *B* размерностью 2x5.

Массив *B*

0					
1					
номера элементов	0	1	2	3	4

Имя массива должно начинаться с буквы и представлять собой непрерывную последовательность букв, цифр и знаков подчеркивания.

Например:

M_1

Massiv_x

B2_5

В качестве имени массива нельзя использовать ключевые слова, такие как *function*, *single* и другие.

Для объявления массива используется команда, которая имеет следующую конструкцию:

DIM <имя_массива> (размерность) **AS** <тип_данных>

Например:

Dim A(5) As Single

При обращении к этой команде в памяти ПК будет выделена область с именем *A*, предназначенная для хранения 5 однотипных элементов объемом 5x2=10 байт (т.к. тип каждой переменной *Single*, см. табл. 1).

Dim B(2,5) As Single

При обращении к этой команде в памяти ПК будет выделена область с именем *B*, предназначенная для хранения таблицы значений 2x5 объемом

$5 \times 5 \times 2 = 50$ байт (т.к. тип каждой переменной *Single*, см. табл. 1).

Для обращения к элементам массива после имени массива в скобках необходимо указать номер элемента. Например, A(2) или B(0,3).

Перед запуском программы необходимо описать все массивы, которые в ней используются!

Ввод и вывод данных

Ввод данных

а) ввод значения переменной в окно ввода

Команда, которая формирует окно ввода информации, имеет следующий вид:

```
<ИМЯ_ВВОДИМОЙ_переменной> = InputBox ("текст в окне ",  
"заголовок окна ", <ИМЯ_ВВОДИМОЙ_переменной>)
```

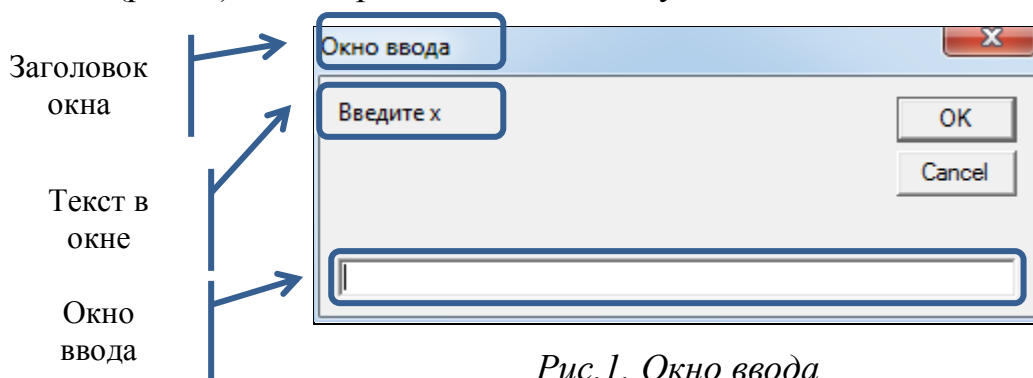
Пример использования команды:

Листинг 1. Фрагмент программы

```
Dim x As single
```

```
x = InputBox("Введите x", "Окно ввода", x)
```

При обращении к этой команде на экране появится диалоговое окно следующего вида (рис. 1). В которое пользователь будет вводить значение переменной *x*.



б) ввод данных из ячейки рабочего листа Excel

Команда, которая позволяет ввести значение переменной из ячейки рабочего листа Excel, имеет вид:

`<имя_вводимой_переменной>=Range ( "адрес ячейки" ) .Value`

Пример использования команды:

Листинг 2. Фрагмент программы

```
Dim x As single
x=Range ("B2") .Value
```

При обращении к этой команде в переменную *x* будет занесено значение из ячейки *B2*.

Вывод данных

а) вывод значений переменной в окно

Команда, которая формирует окно вывода информации, имеет следующий вид:

MsgBox ("текст в окне ")

Пример использования команды:

Листинг 3. Фрагмент программы

```
Dim x As single
Dim y As single
x= InputBox("Введите x", "Окно ввода", x)
y=x^3
MsgBox ("Значение переменной y= "&y)
```

Знак *&* необходим для соединения текстовых и числовых данных.

При обращении к этой команде на экране появится диалоговое окно следующего вида (рис. 2).

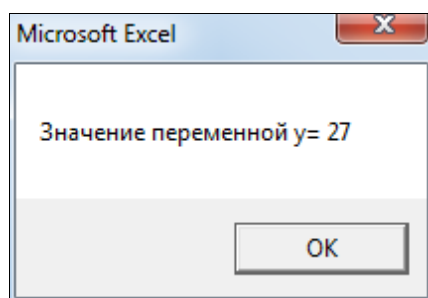


Рис.2. Окно вывода

б) вывод данных в ячейку рабочего листа Excel

Команда, которая позволяет вывести значение переменной в ячейку рабочего листа Excel, имеет вид:

Range ("адрес ячейки") .Value=<имя_переменной>

Пример использования команды:

Листинг 4. Фрагмент программы

```
Dim x as single
Dim y as single
x=Range ("B2").Value
y=x^3
Range ("B3") .Value=y
```

При обращении к этой команде в ячейку B3 будет занесено значение переменной y .

Встроенные функции VBA

Встроенные функции VBA образуют основу, позволяющую программисту не задумываться над реализацией некоторых часто встречающихся стандартных действий.

Математические функции и арифметические операторы

Арифметические операторы	VBA
Сложение (+)	+
Вычитание (-)	-
Умножение (*)	*
Деление (/)	/
x^m	x^m
$\sqrt{x}$	<code>Sqr (x)</code>
$\sqrt[m]{a^n}$	$(a^n)^{(1/m)}$
Остаток от деления числа a на число b	$a \bmod b$
Целая часть от деления числа a на число b	$a \backslash b$

Математическая функция	Функция VBA
$ x $	<code>abs (x)</code>
$\tan(x)$	<code>tan (x)</code>
$\arctg(x)$	<code>atn (x)</code>
$\cos(x)$	<code>cos (x)</code>
$\sin(x)$	<code>sin (x)</code>
$\arccos(x)$	<code>application.acos (x)</code>
$\arcsin(x)$	<code>application.asin (x)</code>
$\ln(x)$	<code>log (x)</code>
$\log_a(b)$	$\log(b) / \log(a)$
e^x	<code>exp (x)</code>
π	<code>4*atn (1)</code>

Функции преобразования данных

`Cstr (число)` – преобразование числовой переменной в текстовую.

`Val (строка)` – возвращает числа, содержащиеся в строке.

Запуск редактора VBA. Окно редактора VBA

Программа, написанная на VBA, имеет модульную структуру.

Модуль – это страница с кодом, относящаяся к приложению.

На рис. 3 представлено окно редактора VBA, в котором открыто два модуля, содержащих тексты программ.

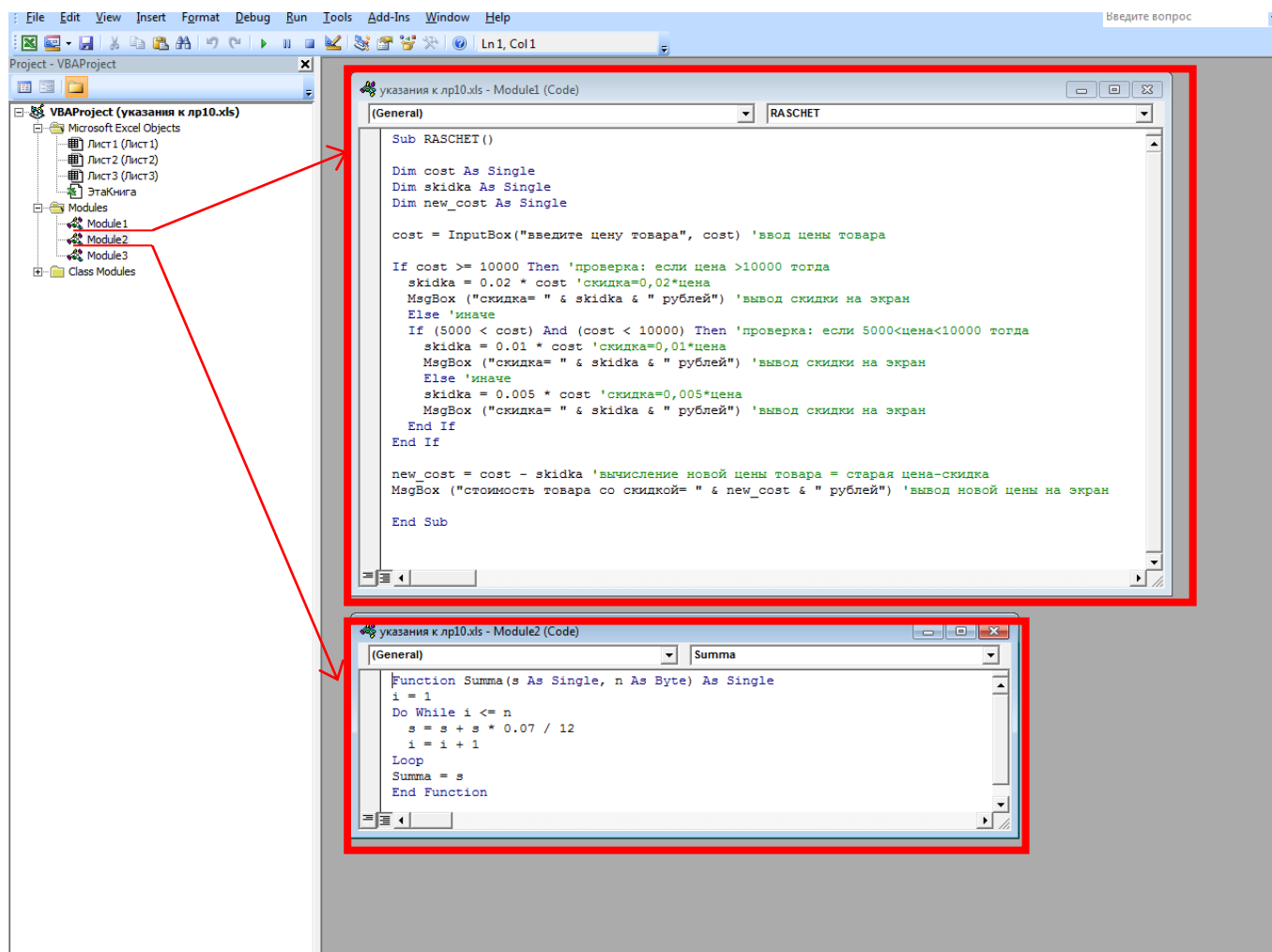
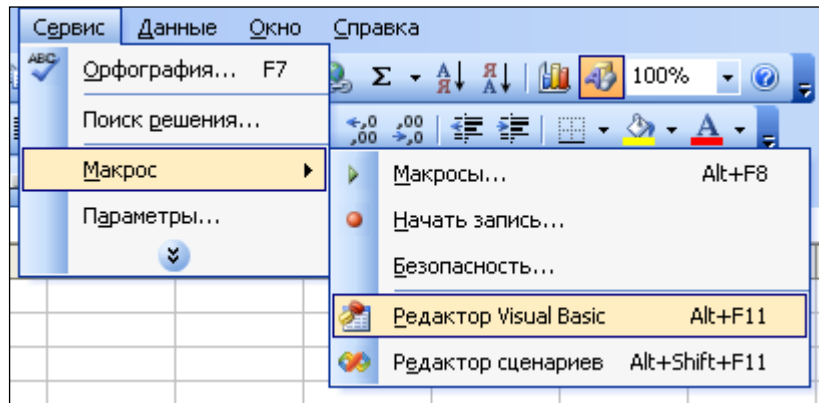


Рис. 3. Модульная структура

Для создания пользователем своего программного модуля необходимо:

1. Открыть окно редактора *Visual Basic* командой *Сервис / Макрос / Редактор Visual Basic* в MS Excel 2003 (рис. 4 а) или нажатием кнопки *Visual Basic* в разделе *Код* на вкладке *Разработчик* в MS Excel 2007 (рис. 4 б).

а)



б)

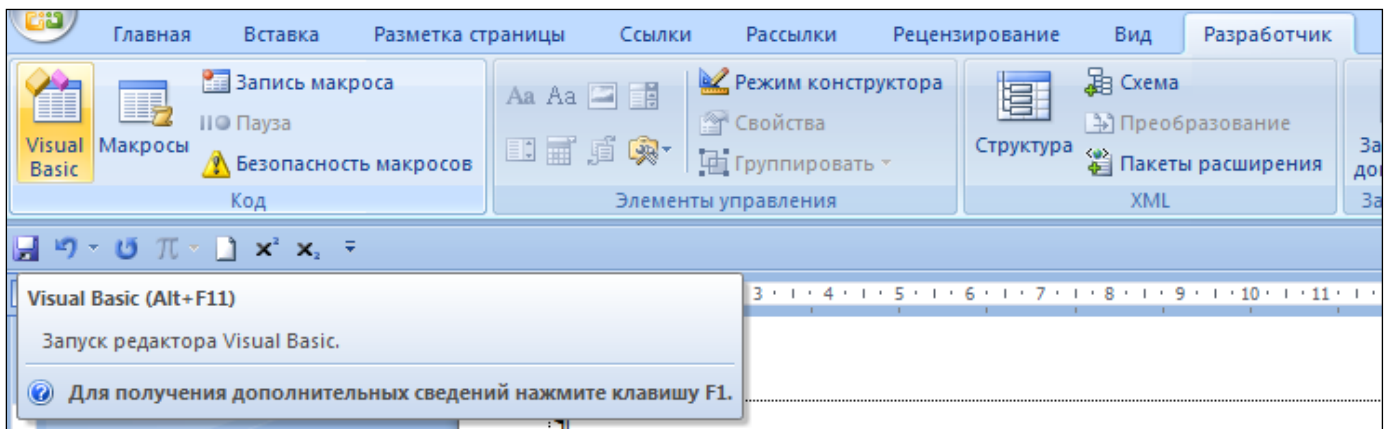


Рис.4. Вызов редактора Visual Basic

а) MS Excel 2003, б) MS Excel 2007

2. В окне редактора *Visual Basic* в пункте главного меню *Insert* (*Вставка*) выбрать команду *Module* (*Модуль*).

В разделе *Project* появляется новая папка *Modules*, содержащая новый модуль *Module1*, предназначенный для написания текста программы (рис. 5)

В каждом модуле могут быть записаны одна или несколько подпрограмм (процедур).

Подпрограмма - поименованная или иным образом идентифицированная часть компьютерной программы, содержащая описание определённого набора действий.

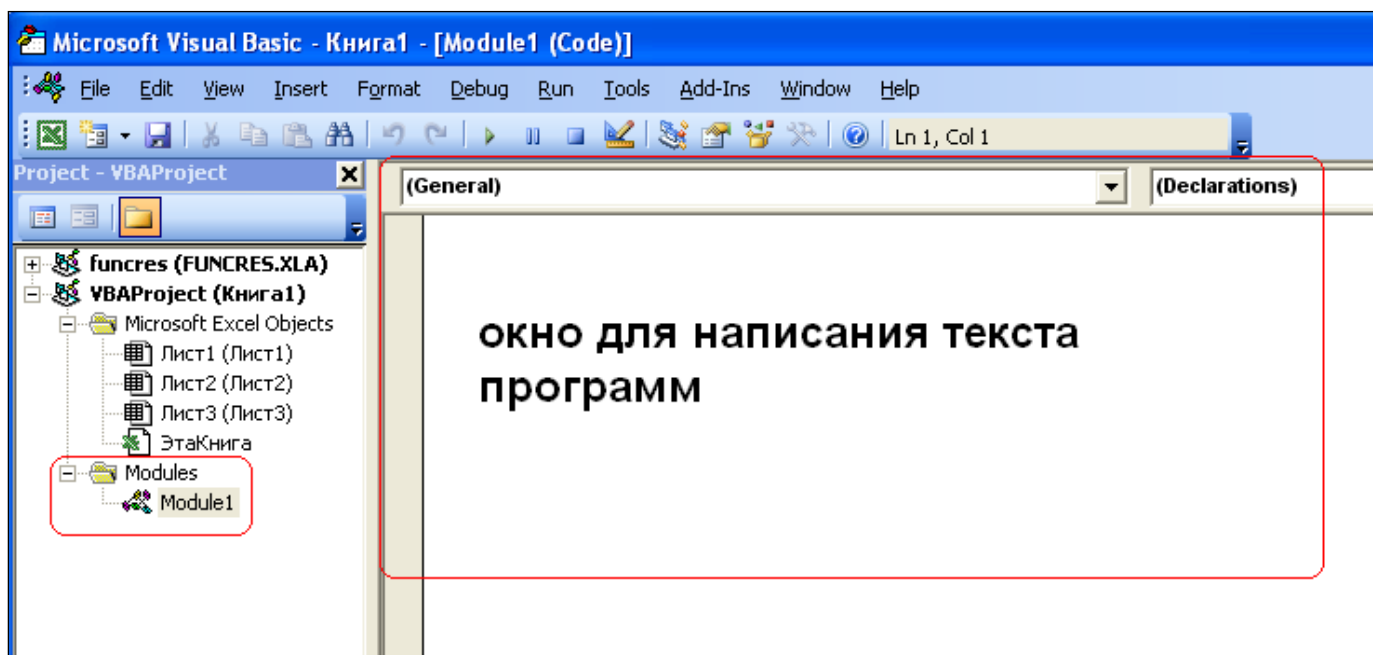


Рис. 5. Рабочее окно редактора

Синтаксис процедуры

Sub <Имя_Процедуры> ()

Тело процедуры

End Sub

Имя_Процедуры – слово, записанное без пробелов, на русском или английском языке.

Тело процедуры – набор операторов, для написания которых, необходимо знание языка *Visual Basic*.

Пример 1.

Задание: написать процедуру для расчета значения функции $y(x) = \sin^2(\pi x) - e^{x-2}$

Порядок работы:

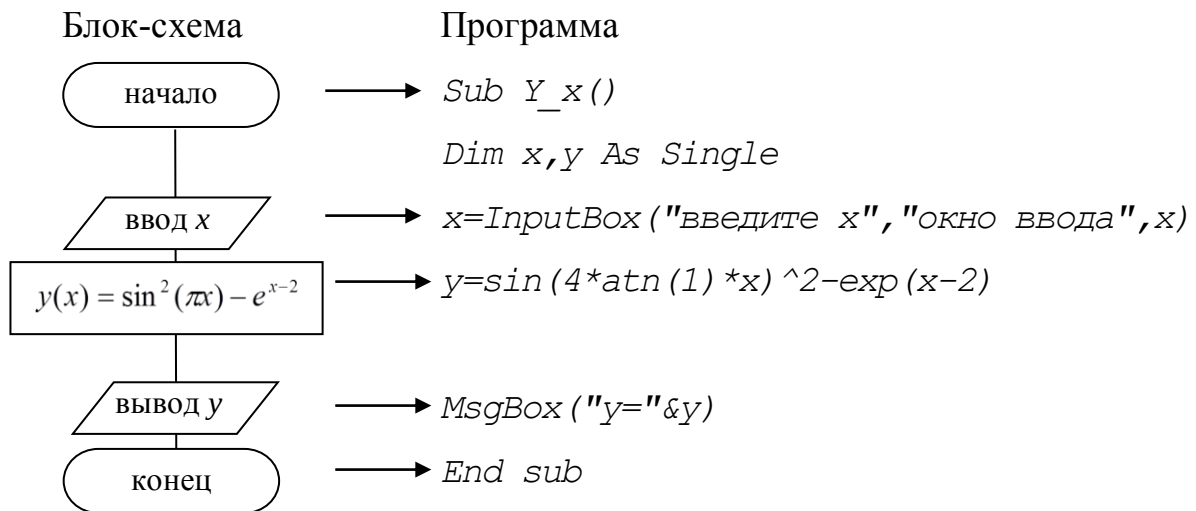
1. Открывается редактор VBA.

MS Office 2003: *Сервис/Макрос/Редактор VBA*

MS Office 2007: вкладка *Разработчик* / кнопка *Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 6).



4. Выполняется тестирование алгоритма.

ВВОД $x=2$
$y(x) = \sin^2(2\pi) - e^{2-2} = 0 - e^0 = 0 - 1 = -1$
печать $y=-1$

5. Выполняется проверка работы программы (рис. 6).

Office 2003: выполняется команда *Сервис / Макрос / Макросы / Y_x / ВЫПОЛНИТЬ*

Office 2007: вкладка *Разработчик* кнопка *Макросы / Макрос Y_x / ВЫПОЛНИТЬ*.

В окно вводится значение переменной x .

В окне вывода появится значение переменной y .

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

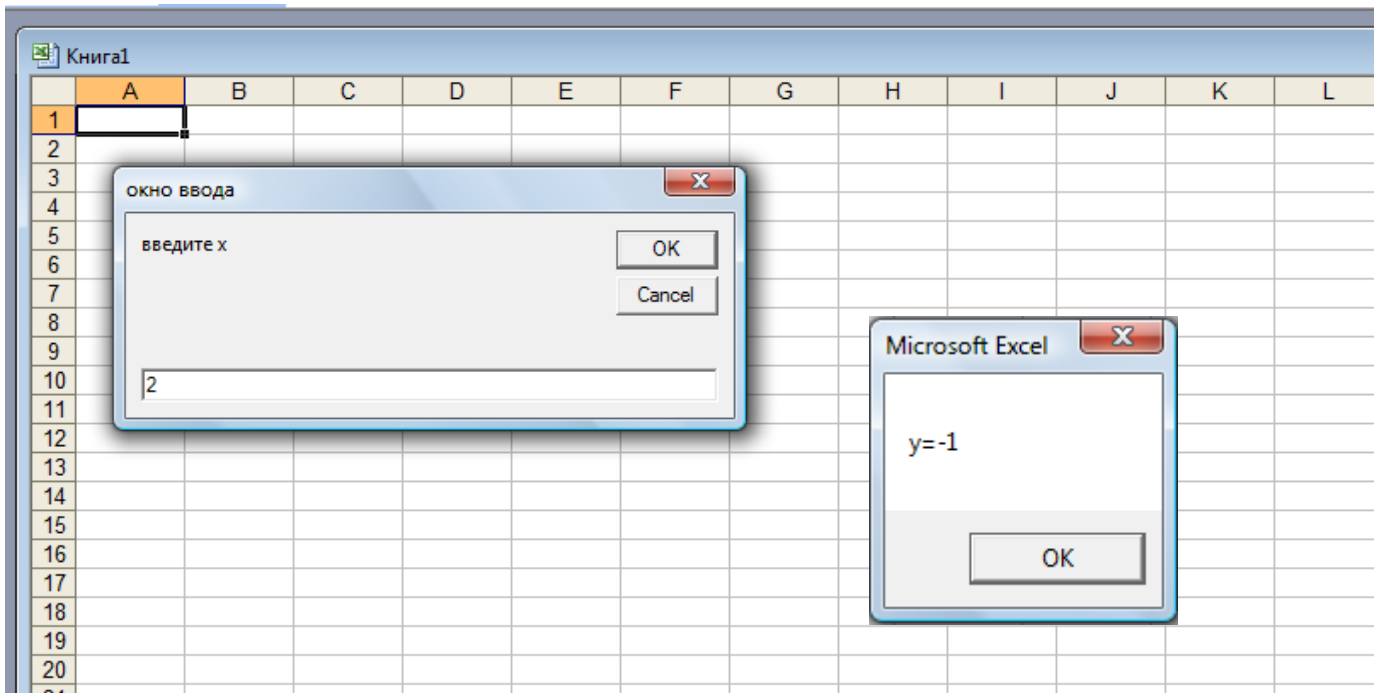


Рис. 6. Реализация программы

Пример 2.

Задание: написать процедуру для расчета значения функции

$$y(x) = \frac{\sin^2(\pi x)}{e} + \sqrt[3]{\cos(x)}$$

Порядок работы:

1. Открывается редактор VBA.

MS Office 2003: *Сервис/Макрос/Редактор VBA*

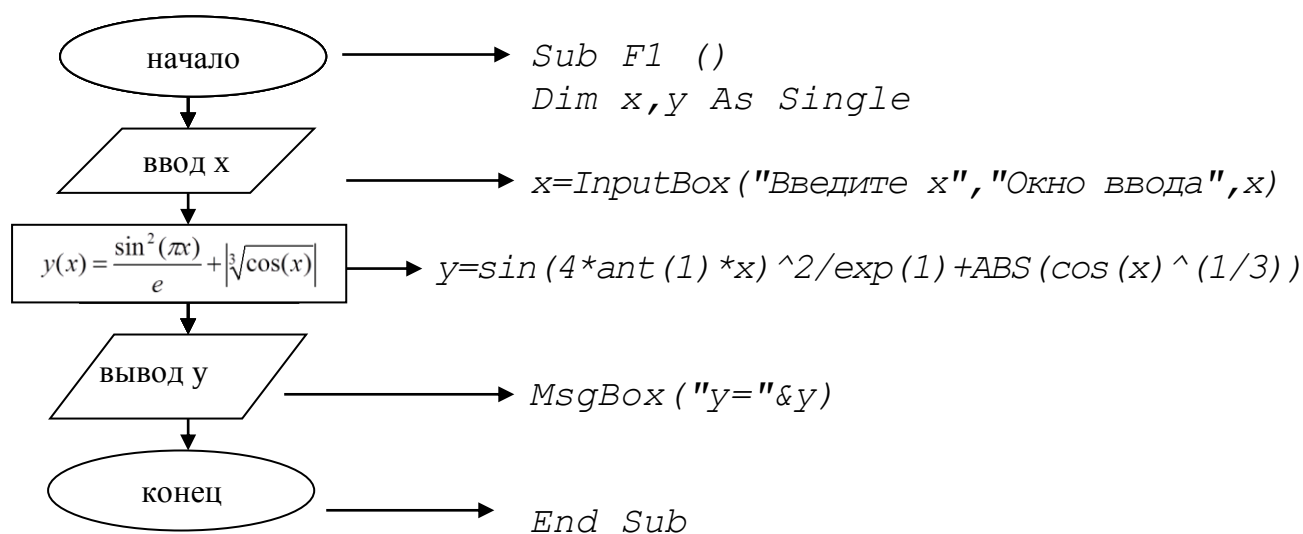
MS Office 2007: вкладка *Разработчик* / кнопка *Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 7).

Блок-схема

Программа



4. Выполняется тестирование алгоритма.

ВВОД $x=2$
$y(x) = \frac{\sin^2(\pi \cdot 0,9)}{e} + \sqrt[3]{ \cos(0,9) } = 1,079$
печать $y=1,079$

5. Выполняется проверка работы программы (рис. 7).

Office 2003: выполняется команда *Сервис / Макрос / Макросы / F1 / ВЫПОЛНИТЬ*

Office 2007: вкладка *Разработчик* кнопка *Макросы / Макрос F1 / ВЫПОЛНИТЬ*.

В окно вводится значение переменной x .

В окне вывода появится значение переменной y .

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

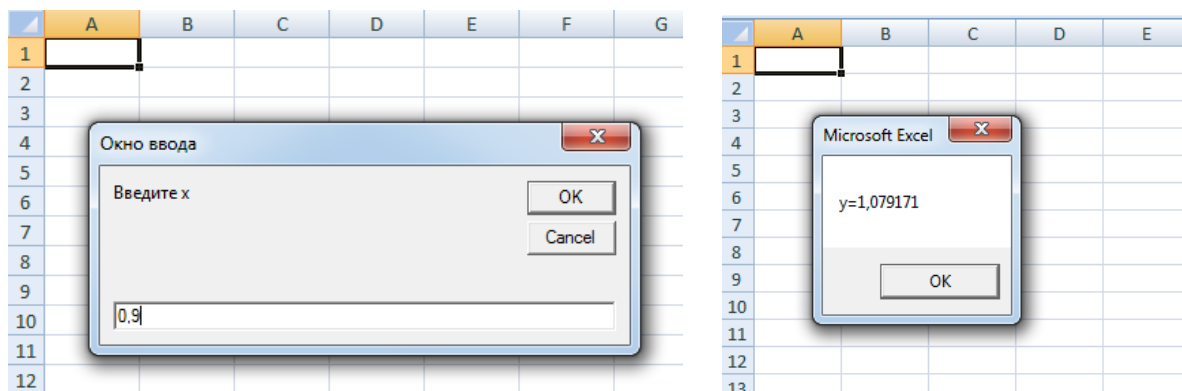


Рис. 7. Реализация программы

Операторы языка VBA

Оператор комментария

Комментарии не влияют на исполнения программ. Они необходимы для пояснения исходных текстов программ.

Оператором комментария в VBA является одинарная кавычка «'».

Комментарий может располагаться на отдельной строке или в конце строки.

Пример использования оператора комментария:

Листинг 5. Фрагмент программы

```
Sub Y_x ()
    'Раздел описания переменных
    Dim y as single,
    Dim n as byte
    'Ввод значения переменной y
    y=Range ("A2").Value
```

Условный оператор

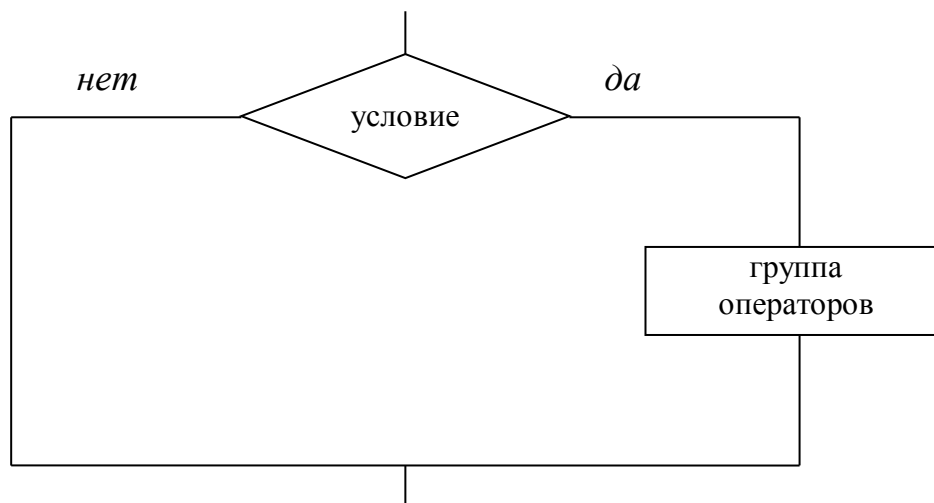
а) неполный оператор

```
IF <условие> THEN
    <группа_операторов>
END IF
```

Порядок работы:

При обращении к оператору проверяется $\langle \text{условие} \rangle$, стоящее после ключевого слова *IF*. Если $\langle \text{условие} \rangle$ истинно, то выполняется $\langle \text{группа_операторов} \rangle$ после ключевого слова *THEN*. Если $\langle \text{условие} \rangle$ ложно, то управление переходит к оператору, идущему после *END IF*.

Фрагмент блок-схемы, соответствующий неполному оператору *IF*.



Пример 3.

Задание: даны два числа k , если $k > m$, то вычислить их разность.

Порядок работы:

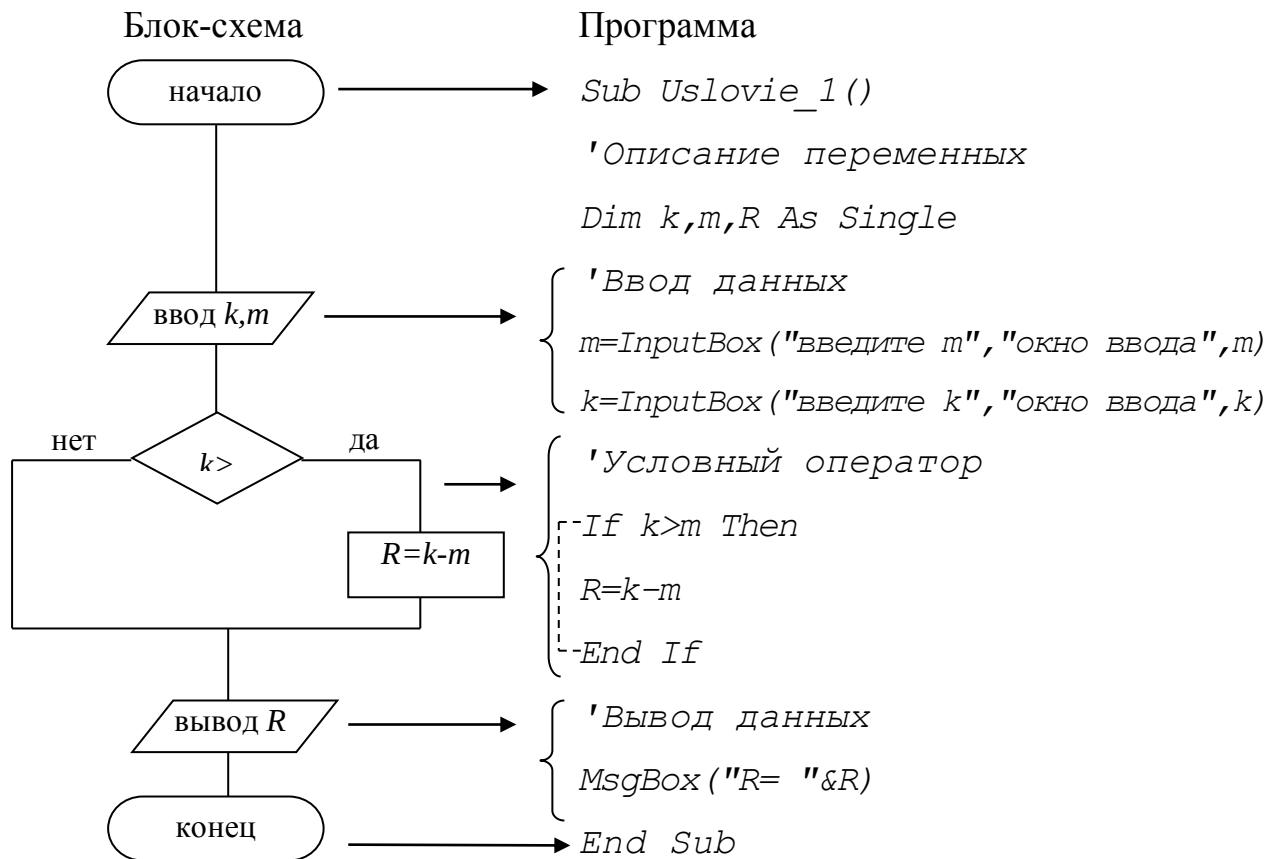
1. Открывается редактор VBA.

MS Office 2003: *Сервис/Макрос/Редактор VBA*

MS Office 2007: вкладка *Разработчик* / кнопка *Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 8).



4. Выполняется тестирование алгоритма.

ввод m=1, k=6
k>m (6>1) условие истинно
R=k-m=6-1=5
печать R=5

5. Выполняется проверка работы программы (рис. 8).

Office 2003: выполняется команда *Сервис / Макрос/ Макросы / Uslovie_1/ Выполнить*

Office 2007: вкладка *Разработчик* кнопка *Макросы/ Макрос Uslovie_1/ Выполнить*.

В окна вводятся значения переменных *k, m*.

В окне вывода появится значение переменной *R*.

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

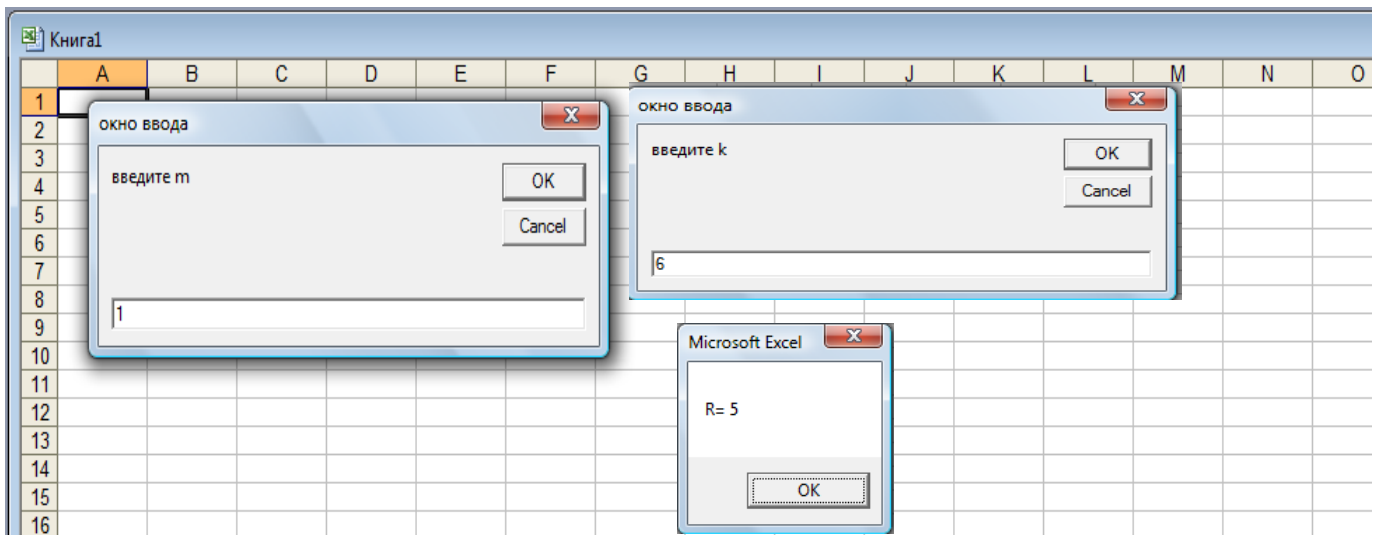
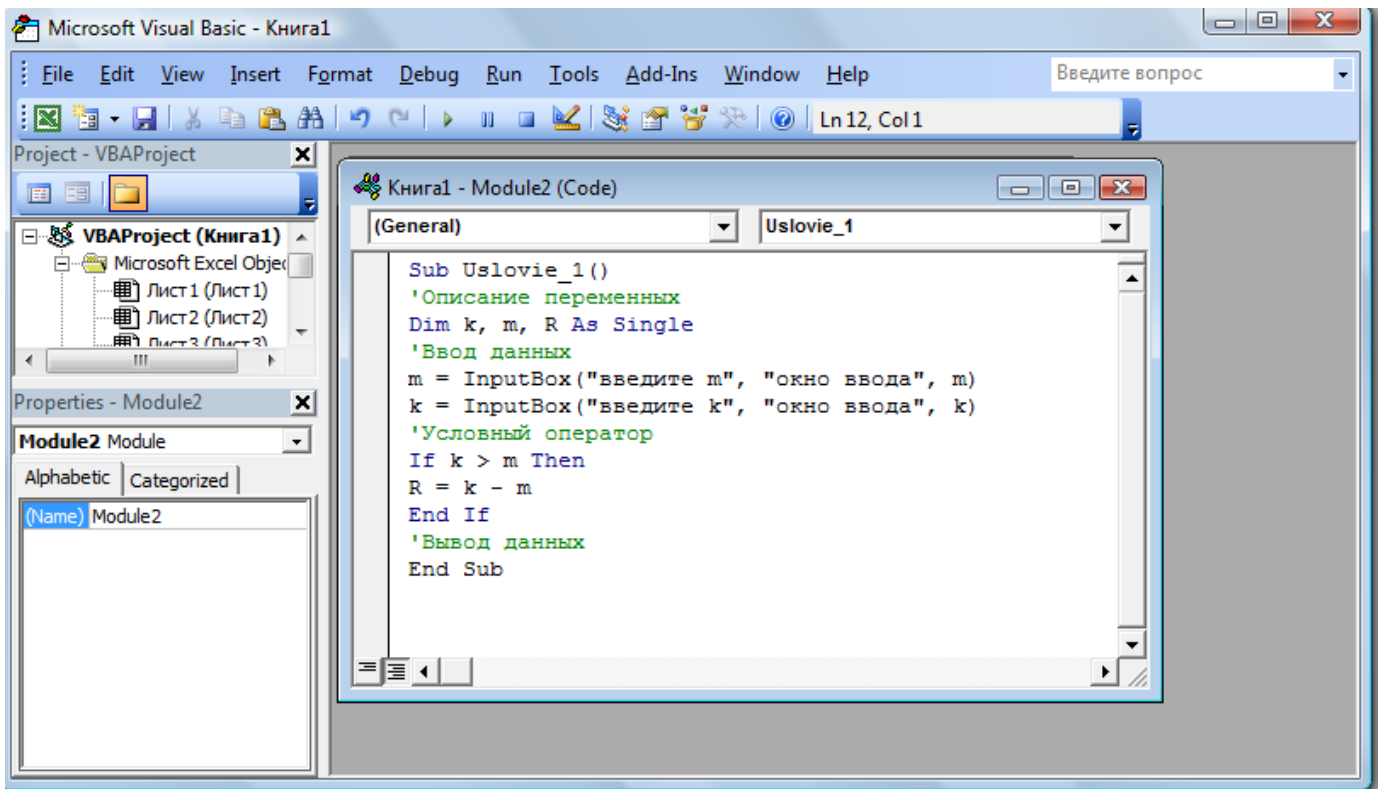


Рис. 8. Реализация программы

б) полный оператор

```

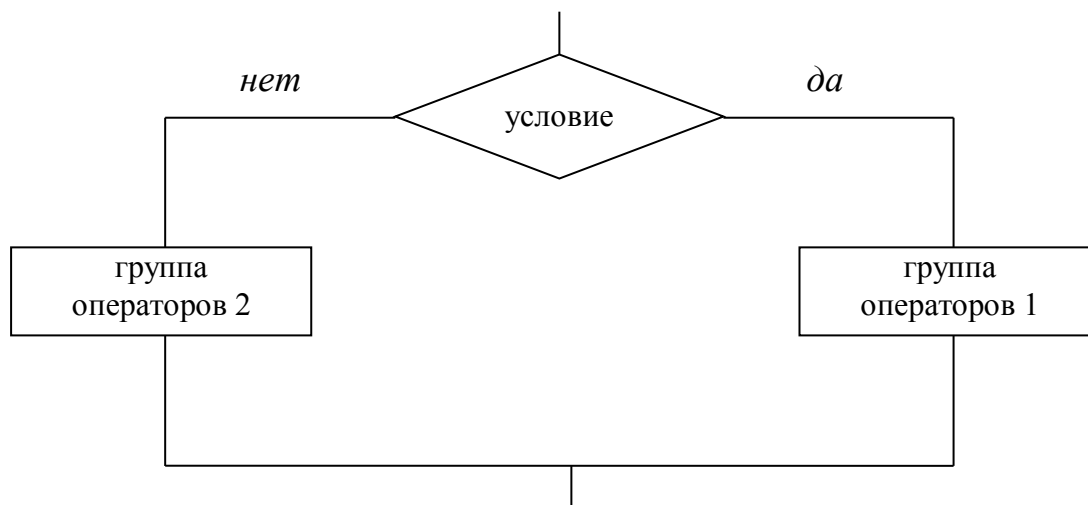
-- IF <условие> THEN
  <группа_операторов_1>
-- ELSE
  <группа_операторов_2>
-- END IF

```

Порядок работы:

При обращении к оператору проверяется $\langle \text{условие} \rangle$, стоящее после ключевого слова *IF*. Если $\langle \text{условие} \rangle$ истинно, то выполняется $\langle \text{группа_операторов_1} \rangle$ после ключевого слова *THEN*. Если $\langle \text{условие} \rangle$ ложно, то выполняется $\langle \text{группа_операторов_2} \rangle$ после ключевого слова *ELSE*.

Фрагмент блок-схемы, соответствующий оператору *IF*.



Пример 4.

Задание: рассчитать значения функции $y = \begin{cases} x^2, & \text{при } x < 2. \\ \sin(x), & \text{при } x \geq 2 \end{cases}$

Порядок работы:

1. Открывается редактор VBA.

MS Office 2003: *Сервис/Макрос/Редактор VBA*

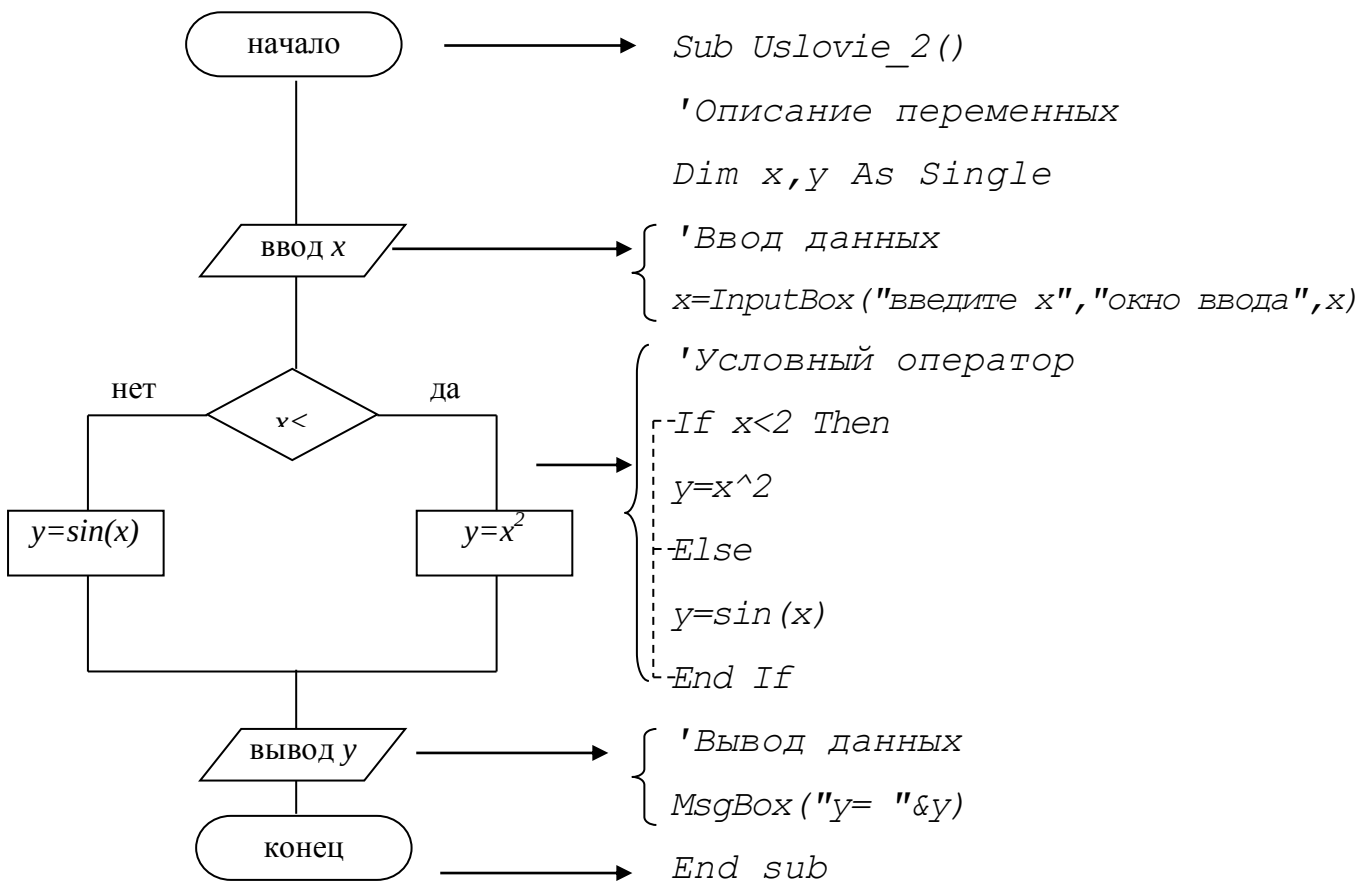
MS Office 2007: вкладка *Разработчик* / кнопка *Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 9).

Блок-схема

Программа



4. Выполняется тестирование алгоритма.

ВВОД x=5
x<2 (5<2) условие ложно
y=sin(x)=sin(5)=-0.95
печать R=5

5. Выполняется проверка работы программы (рис. 9).

Office 2003: выполняется команда Сервис / Макрос/ Макросы / Uslovie_2/ Выполнить

Office 2007: вкладка Разработчик кнопка Макросы/ Макрос Uslovie_2/ Выполнить.

В окно вводится значение переменной x.

В окне вывода появится значение переменной y.

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

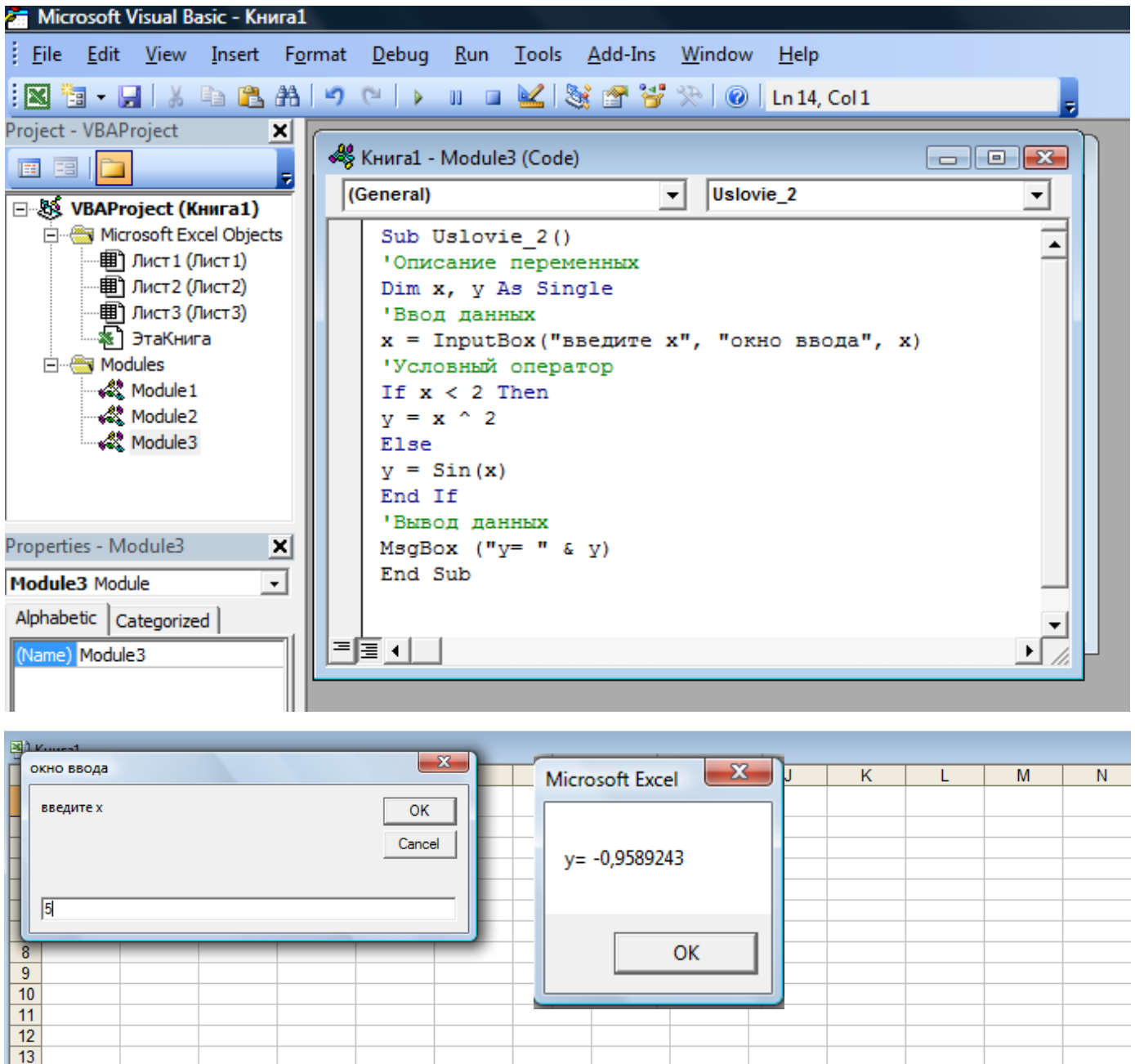


Рис. 9. Реализация программы

Пример 5.

Задание: составить программу для расчета значений функции

$$y(x) = \begin{cases} x^2, & x > 2 \\ \sin^2(x), & -1 \leq x \leq 2 \\ e^x, & x < -1 \end{cases}$$

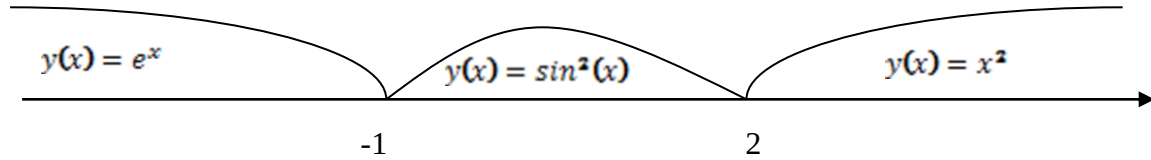
Порядок работы:

1. Открывается редактор VBA.

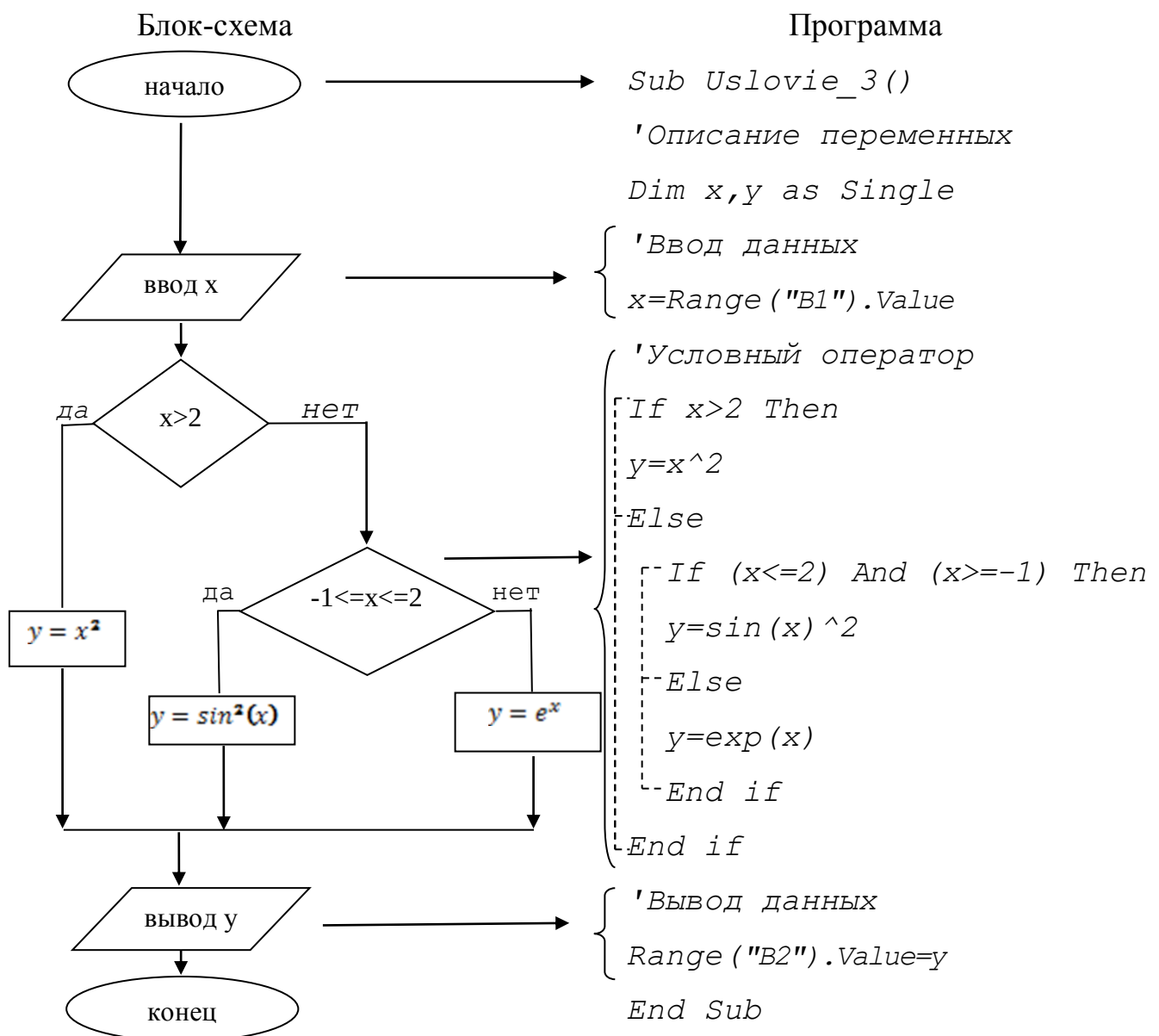
MS Office 2003: *Сервис/Макрос/Редактор VBA*

MS Office 2007: вкладка *Разработчик/кнопка Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).



3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 10).



4. Выполняется тестирование алгоритма.

ВВОД $x=-2$	ВВОД $x=0$	ВВОД $x=7$
$x > 2$ ($-2 > 2$) условие ложно	$x > 2$ ($0 > 2$) условие ложно	$x > 2$ ($7 > 2$) условие истинно
$-1 \leq x \leq 2$ ($-1 \leq -2 \leq 2$) условие ложно	$-1 \leq x \leq 2$ ($-1 \leq 0 \leq 2$) условие истинно	
$y = e^x = e^{-2} = 0,135$	$y = \sin^2 x = \sin^2(0) = 0$	$y = x^2 = 7^2 = 49$
печать $R=0,135$	печать $R=0$	печать $R=49$

5. Выполняется проверка работы программы (рис. 10).

Office 2003: выполняется команда *Сервис / Макрос / Макросы / Uslovie_3 / Выполнить*

Office 2007: вкладка *Разработчик* кнопка *Макросы / Макрос Uslovie_3 / Выполнить*.

В ячейку B1 вводится значение переменной x .

В ячейку B2 выводится значение переменной y .

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

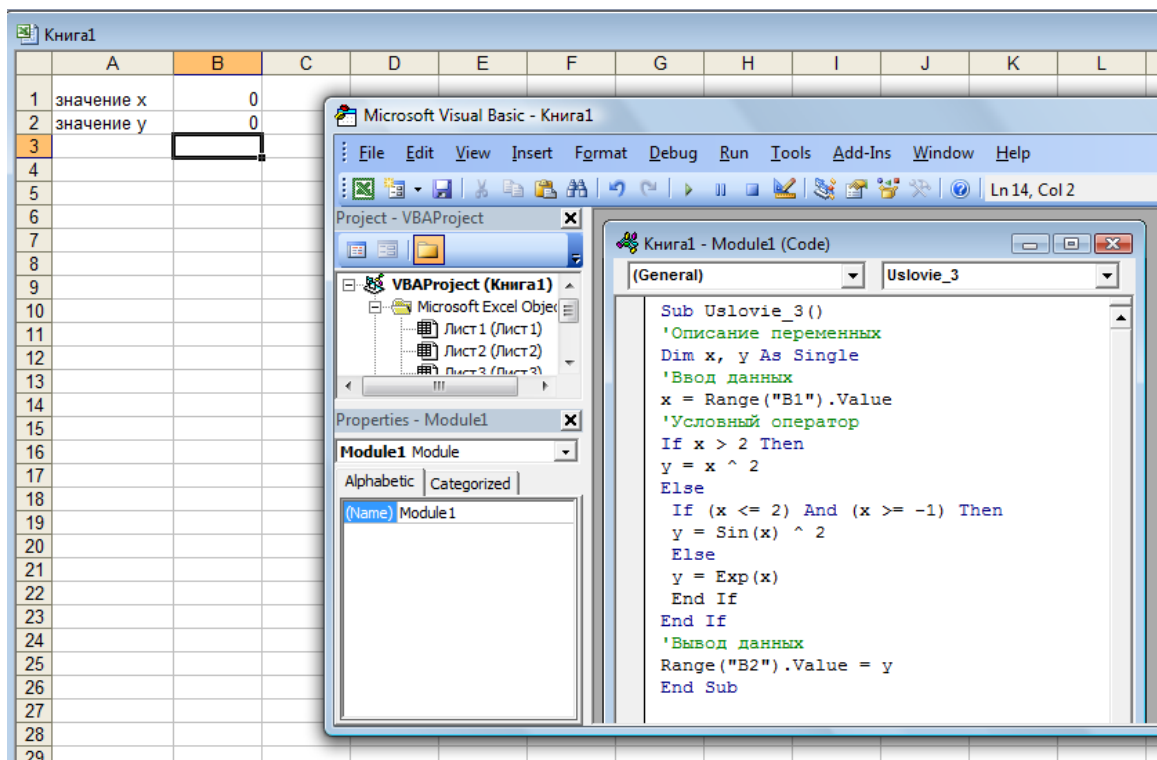


Рис. 10. Реализация программы

Пример 6.

Задание: составить программу для расчета значений функции

$$y(x) = \begin{cases} \sqrt{x-5}, & x > 5 \\ \ln^2(\pi x), & -1 \leq x \leq 2 \\ \frac{1}{x+10}, & x < -3 \end{cases}$$

Порядок работы:

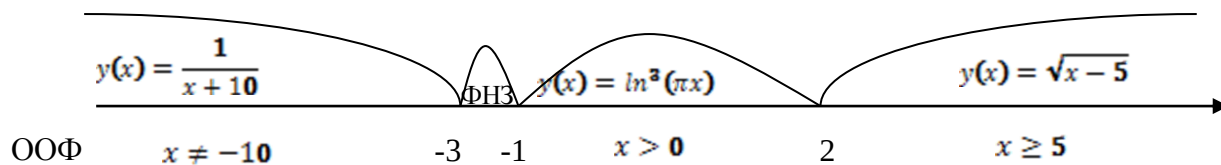
1. Открывается редактор VBA.

MS Office 2003: *Сервис/Макрос/Редактор VBA*

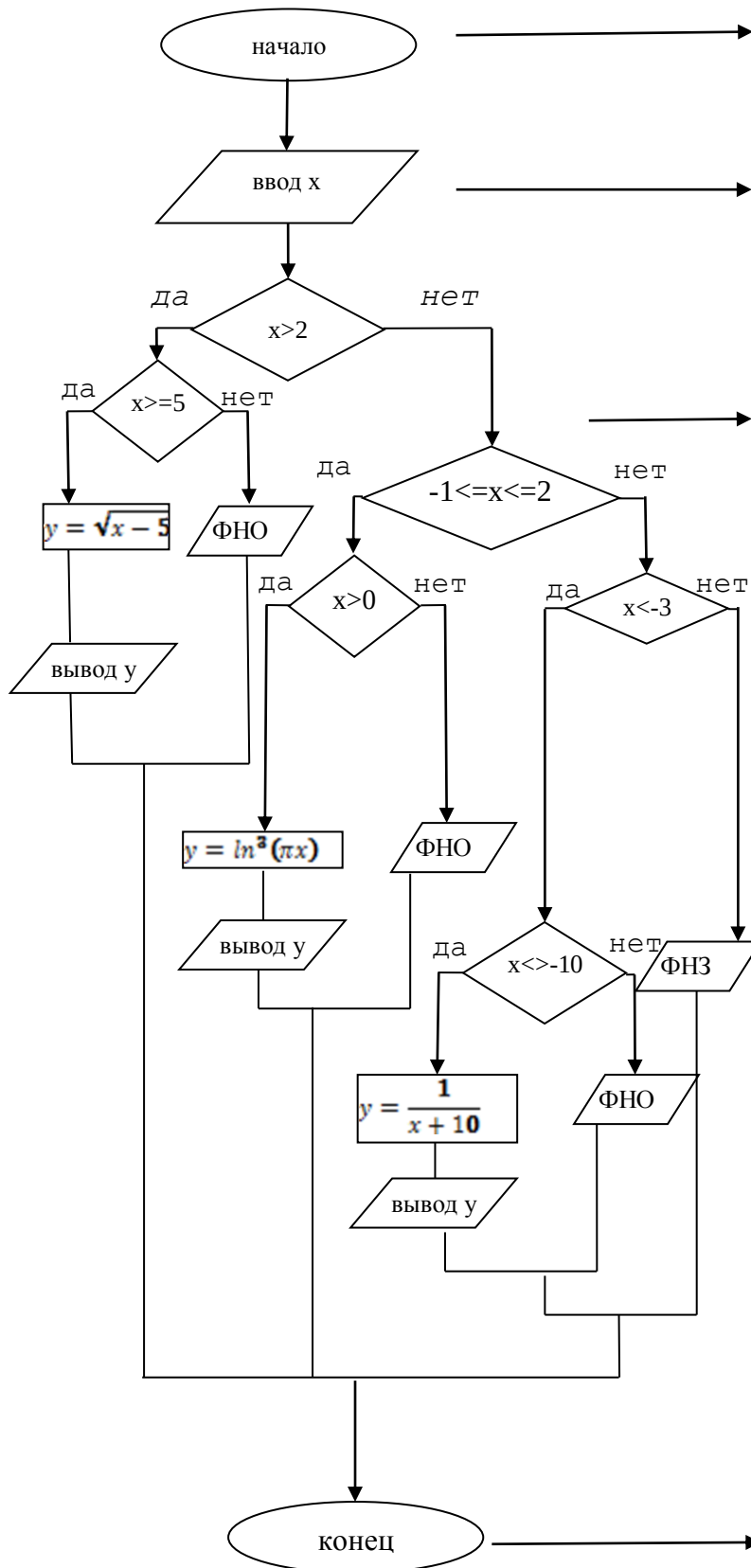
MS Office 2007: вкладка *Разработчик* / кнопка *Visual Basic*.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 11).



Блок-схема



Программа

```

Sub Uslovie_4()
  'Описание переменных
  Dim x,y as Single
  'Ввод данных
  x=Range("B1").Value
  'Условный оператор
  If x>2 Then
    If x>=5 Then
      y=Sqr(x-5)
      Range("B2").Value=y
    Else
      Range("B2").Value="ФНО"
    End If
  Else
    If (x<=2) And (x>=-1) Then
      If x>0 Then
        y=Log(4*atn(1)*x)^3
        Range("B2").Value=y
      Else
        Range("B2").Value="ФНО"
      End If
    Else
      If x<-3 Then
        If x<>-10 Then
          y=1/(x+10)
          Range("B2").Value=y
        Else
          Range("B2").Value="ФНО"
        End If
      Else
        Range("B2").Value="ФНЗ"
      End If
    End if
  End if
End Sub

```

4. Выполняется тестирование алгоритма.

ввод $x=-5$	ввод $x=-2$	ввод $x=1$	ввод $x=7$
$x > 2$ ($-5 > 2$) условие ложно	$x > 2$ ($-2 > 2$) условие ложно	$x > 2$ ($1 > 2$) условие ложно	$x > 2$ ($7 > 2$) условие истинно
$-1 \leq x \leq 2$ ($-1 \leq -5 \leq 2$) условие ложно	$-1 \leq x \leq 2$ ($-1 \leq -2 \leq 2$) условие ложно	$-1 \leq x \leq 2$ ($-1 \leq 1 \leq 2$) условие истинно	
$x \leq -3$ ($-5 \leq -3$) условие истинно	$x \leq -3$ ($-2 \leq -3$) условие ложно		
$x < -10$ ($-5 < -10$) условие истинно			$x \geq 5$ ($7 \geq 5$) условие истинно
$y = 1/(x+10) = 1/(-5+10) = 0,2$		$y = \ln^3(\pi x) = \ln^3(1\pi) = 1,49$	$y = \sqrt{x-5} = \sqrt{7-5} = 1,4$
печать $R=0,2$	Печать ФНЗ	печать $R=1,49$	печать $R=1,4$

5. Выполняется проверка работы программы (рис. 11).

Office 2003: выполняется команда *Сервис / Макрос / Макросы /*

Uslovie_4 / Выполнить

Office 2007: вкладка *Разработчик* кнопка *Макросы / Макрос / Uslovie_4 /*

Выполнить.

В ячейку *B1* вводится значение переменной *x*.

В ячейку *B2* выводится значение переменной *y*.

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел.

[Создание кнопок быстрого вызова](#)).

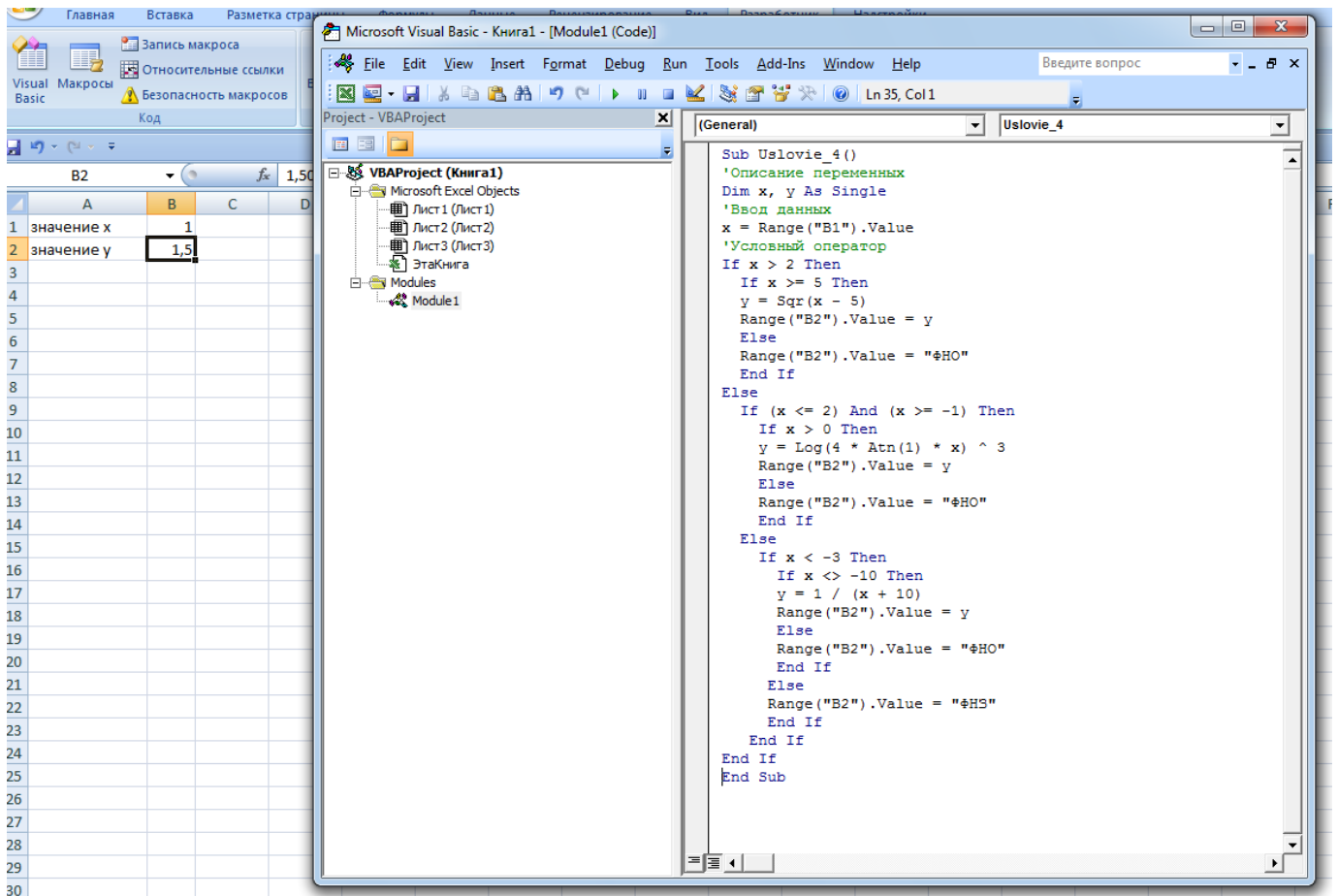


Рис. 11. Реализация программы

Операторы цикла

а) Цикл со счетчиком (For ... Next)

Синтаксис:

```

For <СчетчикЦикла> = <начало> TO <конец> [Step <шаг>]
    <Тело цикла>
Next <СчетчикЦикла>

```

Описание и работа оператора

Цикл со счётчиком *For ... Next* – это цикл, в котором некоторая числовая переменная <СчетчикЦикла> изменяет своё значение от заданного начального значения <начало> до конечного значения <конец> с некоторым шагом, и для каждого значения этой переменной тело цикла выполняется один раз.

Другими словами оператор цикла обеспечивает выполнение *<Тела цикла>* до полного перебора всех значений параметра цикла от начального до конечного с соответствующим шагом.

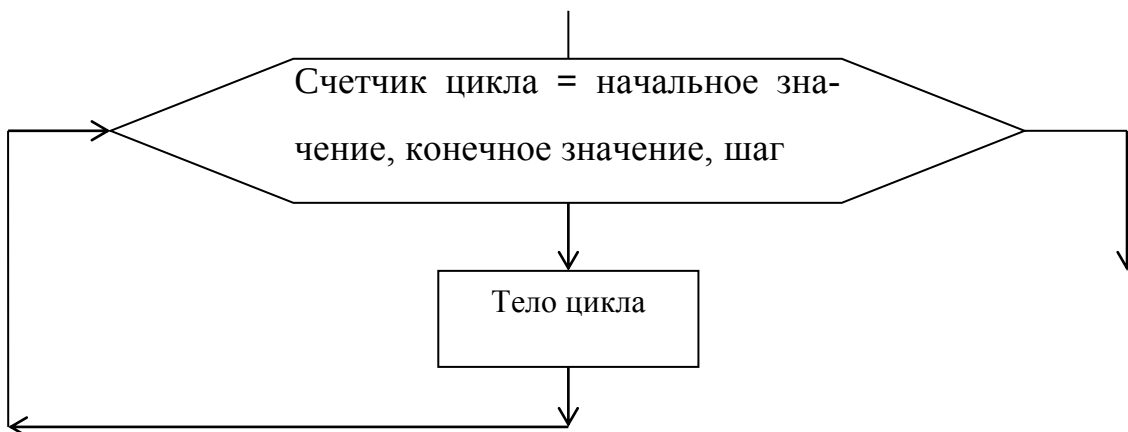
Числовое выражение *<шаг>* не обязательно. Его значение также определяется в начале цикла и прибавляется к счетчику цикла всякий раз, когда завершается выполнение тела цикла и вычисления достигают строки *Next*. Если шаг цикла не указан, то его значение считается равным 1.

<Тело цикла> – это последовательность операторов, которая будет выполнена заданное число раз.

Цикл завершится, когда счетчик цикла станет больше значения *<конец>*.

Цикл может ни выполниться ни разу, если начальное значение счетчика цикла больше его конечного значения.

Фрагмент блок-схемы, соответствующий оператору *For ... Next*.



Пример 7.

Задание: последовательность чисел задается законом:

$$a_1 = 12,$$

$$a_i = 3 + \frac{2}{i} \cos^2(a_{i-1} - 2),$$

$$i=2,3,4\ldots$$

Написать программу для расчета элемента последовательности с номером *M*,

где M вводит пользователь.

Порядок работы:

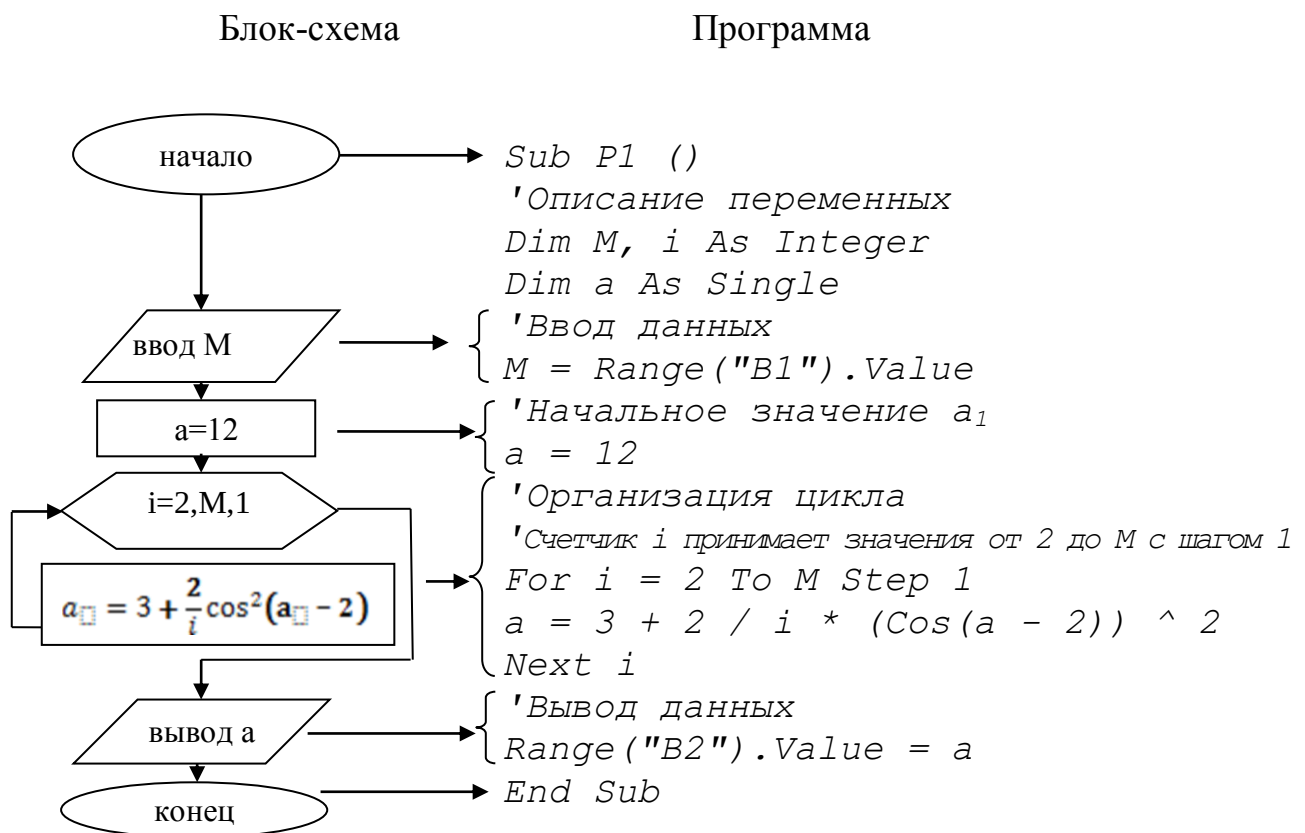
1. Открывается редактор VBA.

Office 2003: Сервис/Макрос/Редактор VBA

Office 2007: вкладка Разработчик / кнопка Visual Basic.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 12).



4. Выполняется тестирование алгоритма.

ввод М=4	
а=12 (первый элемент)	
Значения счетчика i	Значения переменной a
i=2	$a = 3 + 2/2 * \cos^2(12-2) \approx 3.7$
i=3	$a = 3 + 2/3 * \cos^2(3.7-2) \approx 3.01$
i=4	$a = 3 + 2/4 * \cos^2(3.01-2) \approx 3.14$
печать а=3.14	

5. Выполняется проверка работы программы (рис. 12).

В ячейку B1 вводится значение переменной M.

Office 2003: выполняется команда *Сервис / Макрос/ Макросы / P1 / ВЫПОЛНИТЬ*

Office 2007: вкладка *Разработчик* кнопка *Макросы/ Макрос P1 / Выполнить*.

В ячейке B2 появится значение переменной a.

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

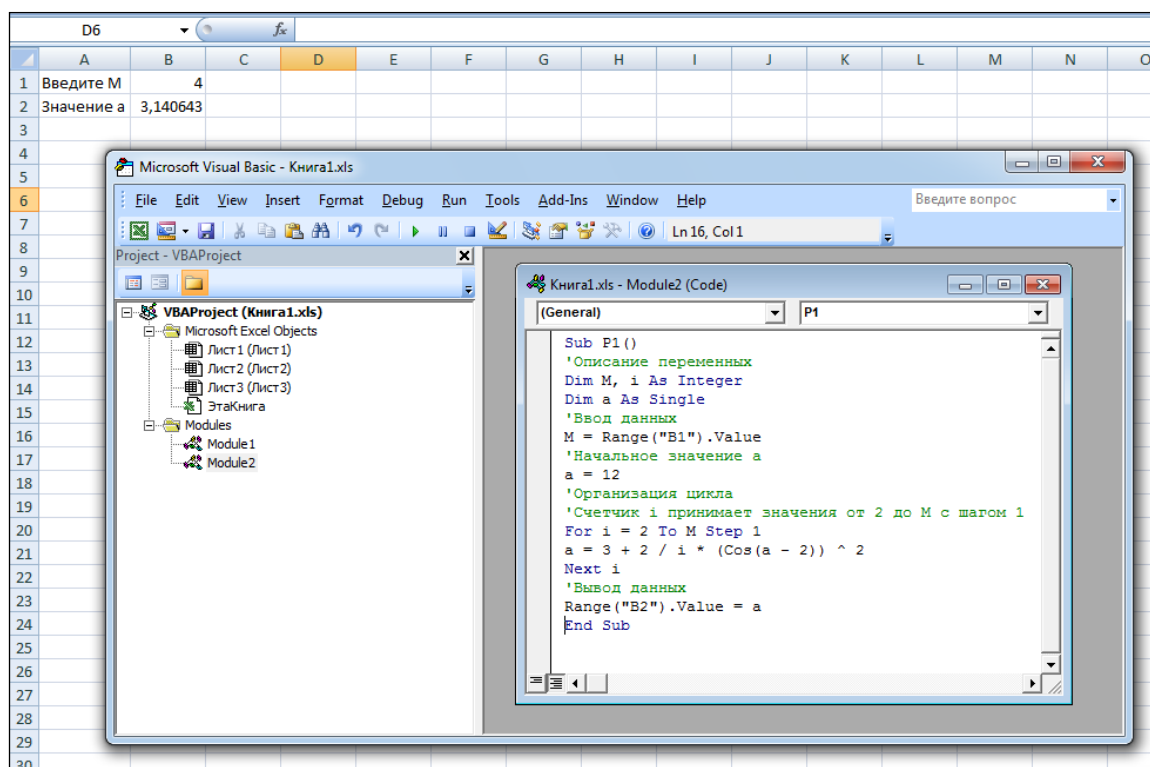


Рис. 12. Реализация программы

б) Цикл с предусловием (*Do While- Loop*)

Синтаксис:

Do While <условие>

<Тело цикла>

Loop

Описание и работа оператора

При обращении к оператору цикла проверяется *<условие>*. Если оно истинно, то выполняются операторы *<Тела цикла>*, если условие ложно, то управление переходит к оператору, стоящему после ключевого слова *Loop*.

<Тело цикла> повторяется, пока заданное условие не станет ложным (или пока условие является истинным).

<Условие> – это числовое или строковое выражение, принимающее значение *True* или *False*.

<Тело цикла> – это последовательность операторов, которая выполняется пока условие остается истинным.

<Тело цикла> должно содержать оператор, который приводит в итоге к ложности условия, в противном случае цикл будет работать бесконечно.

Пример 8.

Задание: последовательность чисел задается законом:

$$a_1 = x,$$

$$a_{i+1} = -a_i * x^2 \left(\frac{2i - 1}{2i + 1} \right)$$

$$i = 1, 2, 3 \dots$$

x – любое действительное число

Написать программу для поиска первого элемента последовательности, для которого выполняется условие $|a_i| < 0.001$.

Порядок работы:

1. Открывается редактор VBA.

Office 2003: Сервис/Макрос/Редактор VBA

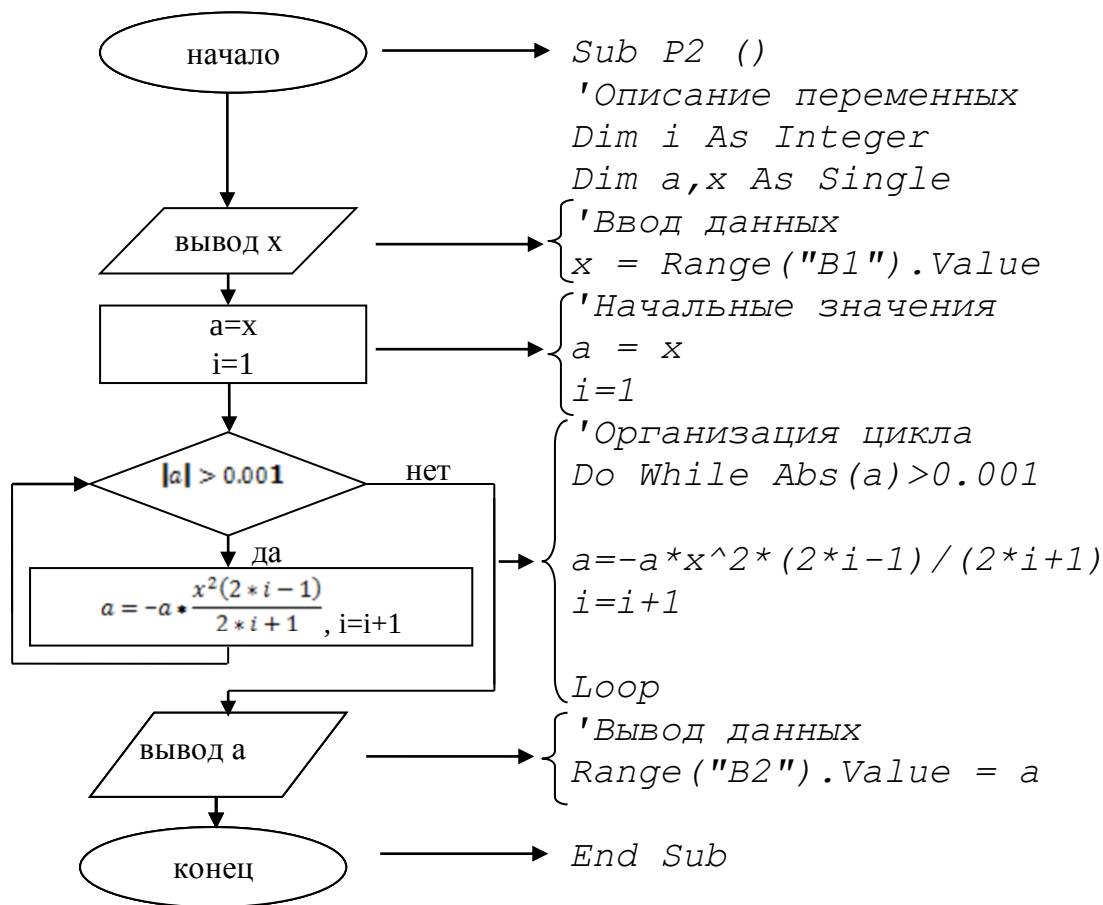
Office 2007: вкладка Разработчик / кнопка Visual Basic.

2. Добавляется новый модуль (*Insert / Module*).

3. Составляется блок-схема. В модуле записывается подпрограмма (рис. 13).

Блок-схема

Программа



4. Выполняется тестирование алгоритма.

ВВОД x=0.3		Условие
a=0.3		$ 0.3 > 0.001$ – условие истинно
i=1	$a = -0,3 * 0,3^2 * (2*1-1) / (2*1+1) \approx -0.009$	$ -0.009 > 0.001$ – условие истинно
i=2	$a = -(-0.009) * 0,3^2 * (2*2-1) / (2*2+1) \approx 0.00048$	$ 0.00048 > 0.001$ – условие ложно
печать a=0.00048		

5. Выполняется проверка работы программы (рис. 13).

В ячейку B1 вводится значение переменной x

Office 2003: выполняется команда Сервис / Макрос / Макросы / P2 /

Выполнить

Office 2007: вкладка Разработчик кнопка Макросы / Макрос P2 / Выполнить.

В ячейке B2 появится значение переменной *a*.

6. Для более быстрого вызова программы удобно создать кнопку (см. раздел. [Создание кнопок быстрого вызова](#)).

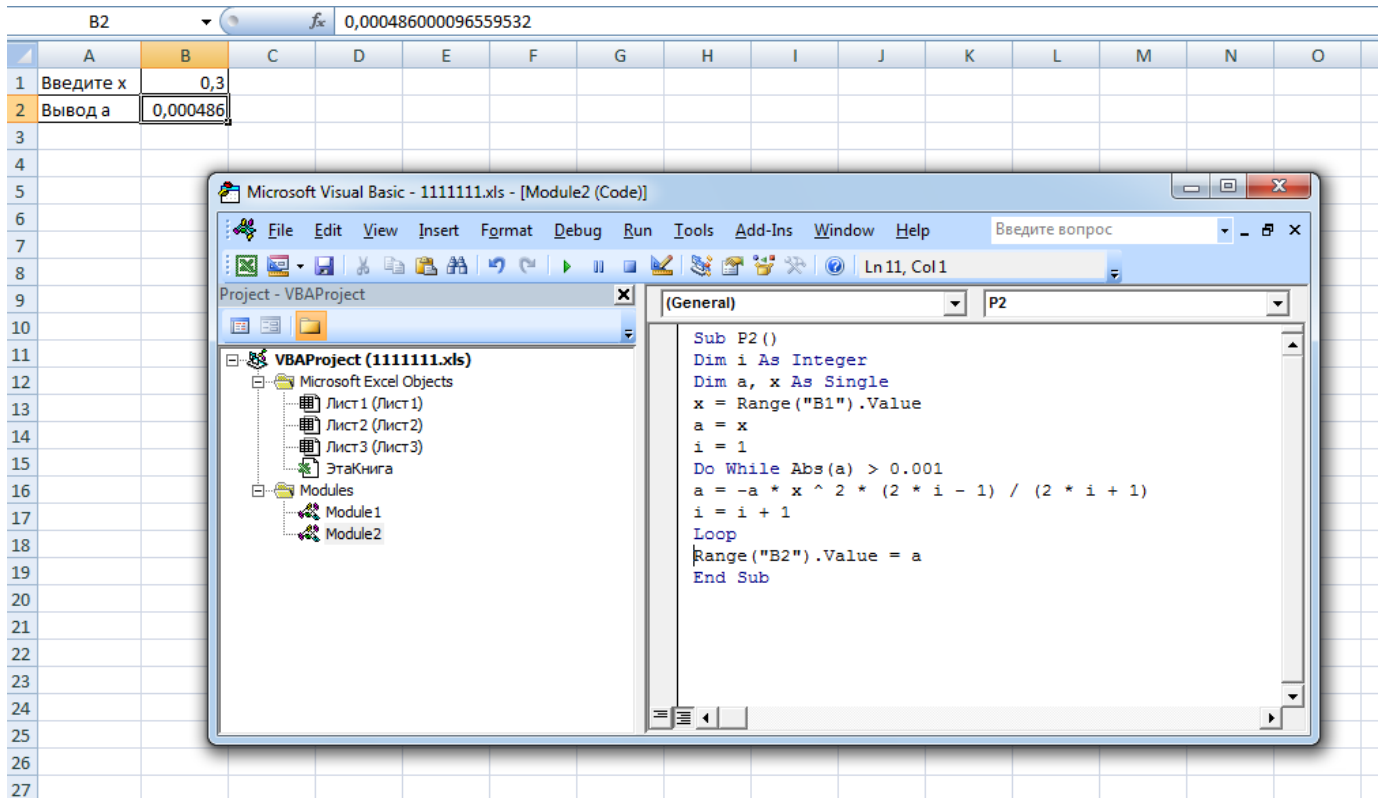


Рис. 13. Реализация программы

Создание кнопок быстрого вызова

Алгоритм:

1. Добавляется кнопка.

Office 2003: выполняется команда Вид / Панели инструментов/ Элементы управления.

На появившейся панели инструментов выбирается объект кнопка и прорисовывается на листе (рис. 14, 16).

Office 2007: на вкладке Разработчик в разделе Элементы управления нажимается кнопка Вставить.

В появившемся меню в разделе Элементы ActiveX выбирается объект кнопка и прорисовывается на листе (рис. 15, 16).

2. По кнопке выполняется щелчок правой кнопкой мыши и в контекстном меню выбирается команда *Объект CommandButton / Edit* (рис. 16).

3. На кнопке записывается текст. Например, *Расчет* (рис. 17).

4. По кнопке выполняется двойной щелчок мышью и в окне редактора VBA записывается имя процедуры, которую будет вызывать кнопка (рис. 18).

5. Закрывается окно редактора VBA.

6. Отключается режим конструктора.

Office 2003: на панели инструментов *Элементы управления* отключается кнопка *Режим конструктора* (🔧).

Office 2007: на вкладке *Разработчик* в разделе *Элементы управления* отключается кнопка *Режим конструктора* (🔧).

7. Теперь для вызова процедуры достаточно нажать кнопку *Расчет*.

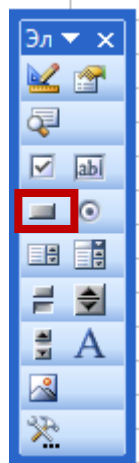


Рис. 14. Панель инструментов *Элементы управления* в MS Excel 2003

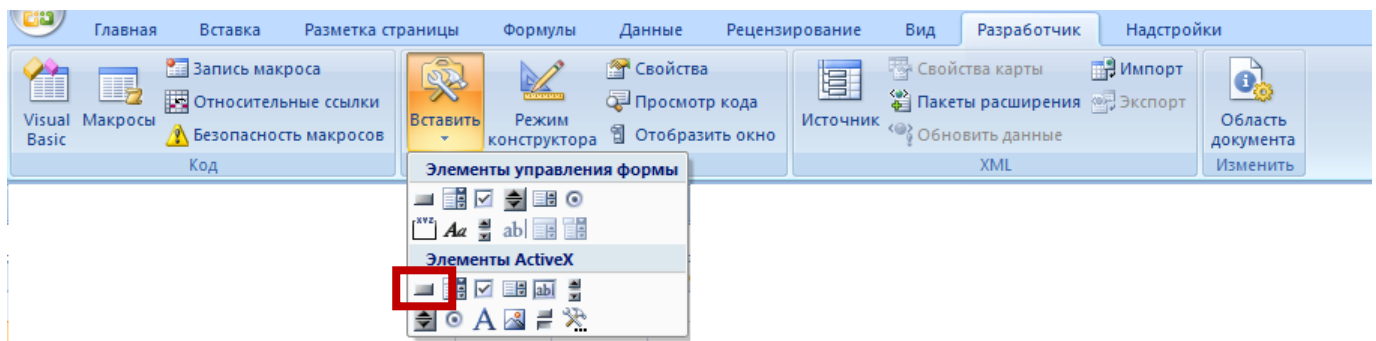


Рис. 15. *Элементы ActiveX* в MS Excel 2007

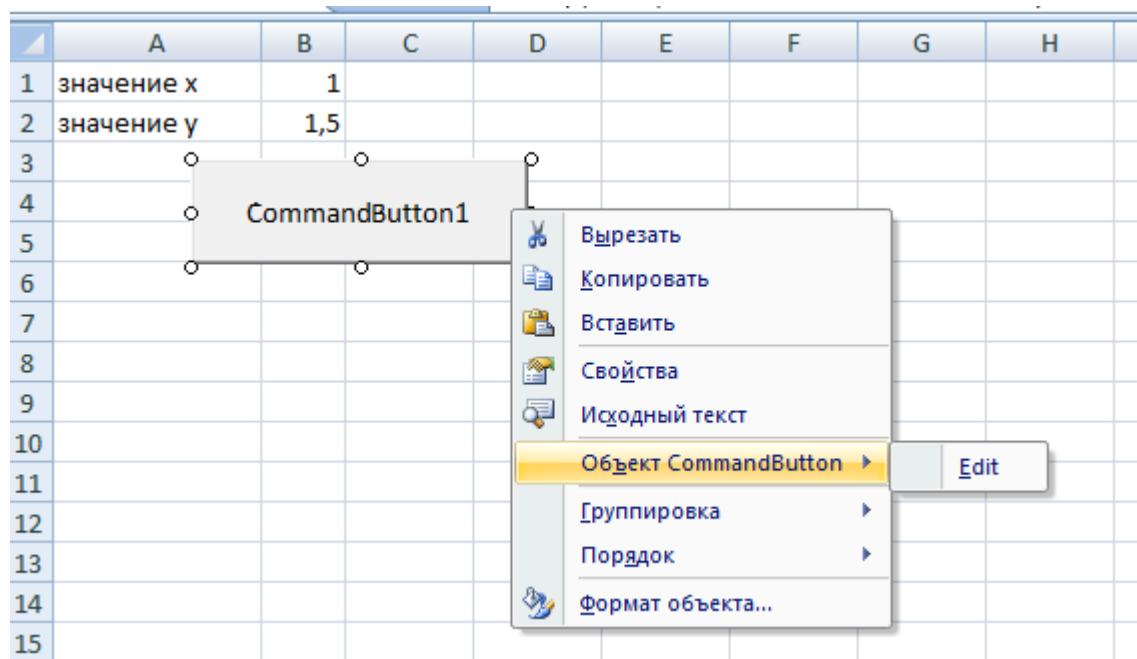


Рис. 16. Добавление кнопки

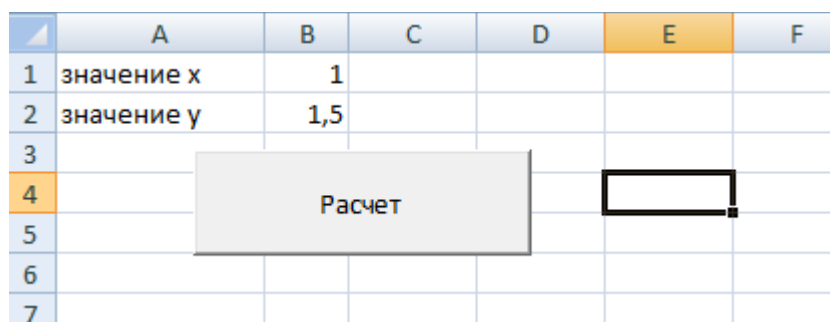


Рис. 17. Подпись на кнопке

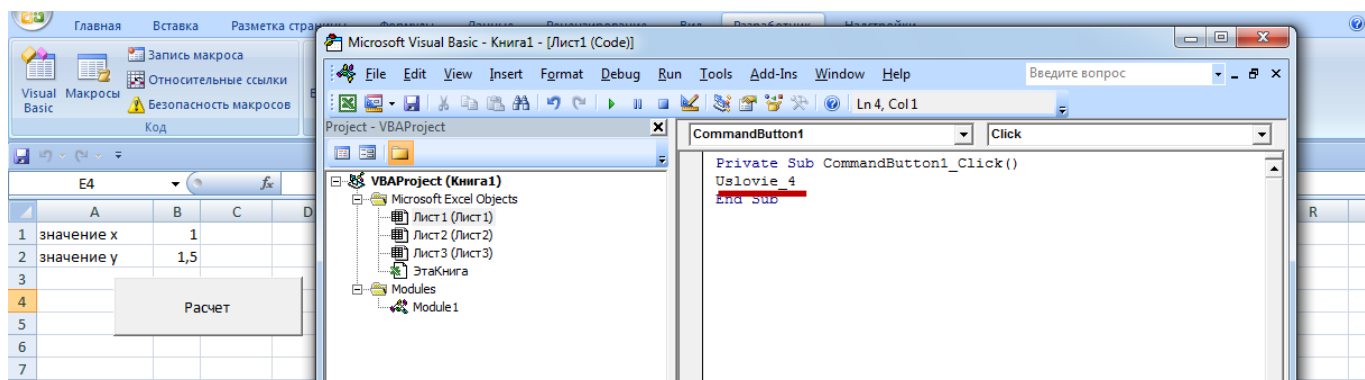


Рис. 18. Запись процедуры

Задания для лабораторных работ

Лабораторная работа «Линейные вычислительные процессы»

Задание: составить блок-схему и написать на VBA программу для расчета значений функции.

Вариант	Функция
1	$y = \cos^2(\pi x) - \ln x - e^x $
2	$y = \sin\left(\frac{\pi x}{2}\right)e^{x-1}$
3	$y = \ln^2 \left \sqrt[3]{x-1} \right + \sin(\pi x)$
4	$y = \cos\left(\frac{e^x - e^{-x}}{2}\right) + \arccos(x)$
5	$y = \cos^2 x^3 - e^x \operatorname{tg}(\pi x)$
6	$y = 2^{\ln^4 x + \sqrt[3]{x-4}}$
7	$y = 3^{\operatorname{tg}(x^2+5x)}$
8	$y = 6e^x + \sqrt[3]{3\cos(\pi x)}$
9	$y = \frac{x+e}{2} \sin(\ln x)$
10	$y = \sin^2(\pi x) + \ln \left x + \sqrt{x^2 + 12} \right $
11	$y = \pi(\arcsin x)^2 - 2\sqrt{ 1-x^2 }$
12	$y = (1+x^2) \sin\left(\frac{\pi x}{2}\right)$
13	$y = \arccos^2(1-x) + \sqrt{ 2-x^2 }$
14	$y = 4 \ln \left \sqrt[3]{x^2} \right - e^{2x}$
15	$y = \sqrt{ 1 + \arccos^2 x } + \cos(\ln x+1)$
16	$y = x + \arcsin(\ln^2 x)$
17	$y = \operatorname{tg}^2\left(\frac{\pi x}{2}\right) + \ln \cos x $

Вариант	Функция
18	$y = \sqrt{\frac{1 - \sin(\pi x)}{1 + \arccos^3 x}}$
19	$y = x\sqrt{4 + x^2} + 3\sin\left(\frac{\pi x}{3}\right)$
20	$y = \cos^2(\pi * (x + 3)) / \operatorname{tg}(x + 2)$
21	$y = 2^{\operatorname{ctg}(x^2 - 3x)}$
22	$y = 2e^{x-7} + \sqrt[3]{3\sin(\pi x)}$
23	$y = \frac{x+e}{3} \ln(\cos x)$
24	$y = \cos^2(\pi x) + \ln\left 2 + \sqrt{x^2 + 12}\right $
25	$y = \pi(\arcsin x)^3 - 3\sqrt{1 - x^3}$

Лабораторная работа «Разветвляющиеся вычислительные процессы»

Задание: составить блок-схему и написать на VBA программу для расчета значений функции.

Вариант	Пример	Вариант	Пример
1	$y = \begin{cases} \frac{\sin(x)}{2} & -12 < x < 6, \\ 2 - x^3, & x = 6, \\ (8 - x), & x > 6. \end{cases}$	11	$y = \begin{cases} \sqrt{ \cos x }, & -6 < x < 6, \\ 0, & x = 6, \\ \sin^2(\pi x) & x > 6. \end{cases}$
2	$y = \begin{cases} \sin^2 x, & -2 < x < 9, \\ \cos(x), & 9 \leq x \leq 19, \\ e^{\frac{1}{x}} & x > 19. \end{cases}$	12	$y = \begin{cases} (1 - x^2), & -8 \leq x \leq 2, \\ \sqrt{ \ln x }, & 2 < x \leq 12, \\ \cos(\pi x), & x > 12. \end{cases}$
3	$y = \begin{cases} \frac{\sin x + \cos x}{2}, & -8 < x \leq 3, \\ \frac{-1}{2e^{x+1}}, & x > 3. \end{cases}$	13	$y = \begin{cases} \sqrt{ \cos x }, & 1 < x < 7,2, \\ (8 + x), & 7,2 \leq x < 10, \\ e^x, & x > 12. \end{cases}$

4	$y = \begin{cases} 0,1 + \cos x, & -10 \leq x < 2, \\ \sin(\pi x), & x = 2, \\ \ln(x), & x > 2. \end{cases}$	14	$y = \begin{cases} \sqrt{ x/4 }, & -10 < x < 5, \\ 3 - x^3, & x = 5, \\ (6 + x), & x > 5. \end{cases}$
5	$y = \begin{cases} e^a a^3 & -10 \leq a \leq 1, \\ \cos(a), & 1 < a < 6, \\ (5 - a), & a \geq 6. \end{cases}$	15	$y = \begin{cases} \sin(x) + \cos(x), & -8 < x \leq 3, \\ 1/(10 + x), & x > 3. \end{cases}$
6	$y = \begin{cases} \cos(x/2), & -18 \leq x < 2, \\ \cos x + \sin x, & 2 \leq x \leq 8, \\ (18 + x), & x > 8. \end{cases}$	16	$y = \begin{cases} \sqrt{ \cos x } + e^x, & x > 10, \\ 1/(3 - x), & x < -10. \end{cases}$
7	$y = \begin{cases} 18 + x & -10 \leq x \leq 3, \\ 1/(20 + x), & x > 3. \end{cases}$	17	$y = \begin{cases} \sqrt{ 8 + x }, & -10 \leq x \leq 3, \\ 1/(20 + x), & x > 3. \end{cases}$
8	$y = \begin{cases} \ln(10 + x), & -3 \leq x \leq 3, \\ e^{x+1}, & 3 < x \leq 12, \\ \cos^2 x, & x > 12. \end{cases}$	18	$y = \begin{cases} \sin x, & -2 < x < 9, \\ \ln^2(x), & 9 \leq x \leq 19, \\ 5x, & x > 19. \end{cases}$
9	$y = \begin{cases} \cos(x) + e^x, & x > 1, \\ 1/(3 + x), & x < -5. \end{cases}$	19	$y = \begin{cases} \cos(\pi x) + e^x, & x > 2, \\ 1/(3 - x), & x < -2. \end{cases}$
10	$y = \begin{cases} \cos x & -15 < x < 0,2, \\ 0,5 \sin x, & 0,2 \leq x \leq 15, \\ \ln(18 + x), & x > 15. \end{cases}$	20	$y = \begin{cases} \cos x, & -15 < x < 0,2, \\ 0,5e^{x+6}, & 0,2 \leq x \leq 15, \\ \ln(4 + x), & x > 15. \end{cases}$

Лабораторная работа «Циклические вычислительные процессы»

Задание 1: составить блок-схему и написать на VBA программу для поиска элемента последовательности с номером M .

Задание 2: составить блок-схему и написать на VBA программу для поиска первого элемента последовательности такого, что выполняется условие $|a_i| < 0.001$. Вывести на экран сам элемент, а также его номер.

Вариант	Последовательность
1	$a_1 = 0,5$ $a_{n+1} = a_n \frac{0,5n}{n+2}$ $n = 1, 2, 3 \dots$
2	$a_1 = 0,3$ $a_{n+1} = a_n \frac{0,3n}{n+1}$ $n = 1, 2, 3 \dots$

Вариант	Последовательность
3	$a_1=0,2$ $a_{n+1} = a_n \frac{0,04 * n}{n + 1}$ $n=1,2,3...$
4	$a_1=0,1$ $a_{n+1} = a_n * 0,1 * \frac{2n - 1}{2n + 1}$ $n=1,2,3...$
5	$a_1=0,2$ $a_{n+1} = -a_n * 0,4 * \frac{2n - 1}{2n + 1}$ $n=1,2,3...$
6	$a_1=0,5$ $a_{n+1} = -a_n * 0,1 * \frac{2n - 1}{2n + 1}$ $n=1,2,3...$
7	$a_1=0,2$ $a_{n+1} = -a_n \frac{0,2 * n}{2n + 1}$ $n=1,2,3...$
8	$a_1=0,1$ $a_{n+1} = a_n \frac{0,2 * (2n - 1)}{2n + 1}$ $n=1,2,3...$
9	$a_1 = 0,1$ $a_{n+1} = a_n \frac{-0,9 * (n + 1)}{2n(n + 2)}$ $n=1,2,3...$
10	$a_0 = 0,5$ $a_{n+1} = -a_n * 0,3 * \frac{2n + 1}{2n + 3}$ $n=1,2,3...$
11	$a_0 = 0,1$ $a_{n+1} = -a_n * 0,1 * \frac{2n + 4}{2n + 3}$ $n=1,2,3...$
12	$a_1 = 0,2$ $a_{n+1} = -a_n * 0,1 * \frac{n}{n + 1}$ $n=1,2,3...$
13	$a_1 = 0,1$ $a_{n+1} = \frac{a_n}{n}$ $n=1,2,3...$
14	$a_1 = 0,1$ $a_{n+1} = -a_n * 0,5 * \frac{(n + 1)}{(n + 2)}$ $n=1,2,3...$

Вариант	Последовательность
15	$a_1 = 0,2$ $a_{n+1} = a_n \frac{0,2}{n+1}$ $n=1,2,3...$
16	$a_1 = 0,1$ $a_{n+1} = a_n \frac{0,1}{(2n+1)}$ $n=1,2,3...$
17	$a_1 = 0,2$ $a_{n+1} = -a_n \frac{0,4}{(2n+2)(2n+1)}$ $n=1,2,3...$
18	$a_1 = 0,1$ $a_{n+1} = -a_n \frac{0,2}{(n+1)}$ $n=1,2,3...$
19	$a_1 = 0,5$ $a_{n+1} = -a_n \frac{0,5}{(2n+2)(2n+1)}$ $n=1,2,3...$
20	$a_1 = 0,7$ $a_{n+1} = -a_n \frac{0,3}{(n+1)}$ $n=1,2,3...$
21	$a_1 = 0,1$ $a_{n+1} = -a_n * 0,1 * \frac{n}{n+1}$ $n=1,2,3...$
22	$a_1 = 0,2$ $a_{n+1} = -a_n * 0,2 * \frac{2 * n}{n+1}$ $n=1,2,3...$
23	$a_1 = 0,1$ $a_{n+1} = -a_n \frac{0,1 * n}{n+2}$ $n=1,2,3...$
24	$a_1 = 0,2$ $a_{n+1} = a_n \frac{0,4}{5n}$ $n=1,2,3...$
25	$a_1 = 0,1$ $a_{n+1} = a_n \frac{0,1 * n}{n+1}$ $n=1,2,3...$