



Т.С. Крайнова
Е.В. Анянова
М.П. Воронов

**ИНФОРМАТИКА
И ПРОГРАММИРОВАНИЕ**
VBA в MS Excel

Екатеринбург
2016

Электронный архив УГЛТУ

МИНОБРНАУКИ РОССИИ

ФГБОУ ВО «УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЛЕСОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра информационных технологий и моделирования

Т.С. Крайнова
Е.В. Анянова
М.П. Воронов

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

VBA в MS Excel

Учебно-методическое пособие
к выполнению лабораторно-практического цикла
для обучающихся по направлению подготовки
09.03.03 «Прикладная информатика»
всех форм обучения

Екатеринбург
2016

Печатается по рекомендации методической комиссии ИЭУ.
Протокол № 2 от 23 сентября 2015 г.

Рецензент – Г.Л. Нохрина, старший преподаватель кафедры ИТиМ.

Редактор А.Л. Ленская
Оператор компьютерной верстки Т.В. Упова

Подписано в печать 16.08.16

Плоская печать

Заказ №

Формат 60×84 1/16

Печ. л. 1,63

Поз. 86

Тираж 10 экз.

Цена руб. коп.

Редакционно-издательский отдел УГЛТУ
Отдел оперативной полиграфии УГЛТУ

ВВЕДЕНИЕ

Microsoft Excel – это программа по работе с электронными таблицами, предназначенными для анализа и отображения данных. В MS Excel можно хранить числовые и текстовые данные, проводить вычисления с данными и создавать диаграммы, основанные на данных.

MS Excel можно представить как тетрадь, содержащую множество страниц, каждая из которых разграфлена на строки и столбцы. Строки обозначаются цифрами, а столбцы – буквами латинского алфавита.

На пересечении строк и столбцов находятся ячейки, каждая из которых может содержать число, текст или формулу. Формулы позволяют выполнять практически любые вычисления на основе данных, содержащихся в других ячейках рабочего листа или других рабочих листов той же или другой рабочей книги.

Приложение MS Excel предоставляет пользователю эффективные средства построения диаграмм для графического представления данных. Когда данные на рабочем листе изменяются, результаты вычисления всех формул автоматически пересчитываются и изменяется вид уже созданных диаграмм.

Программные компоненты рабочей книги (модули, процедуры, формы) объединяются в проект, который сохраняется на диске вместе с документом. Для разработки программ используется редактор Visual Basic.

Visual Basic for Application – это язык программирования, встроенный во множество отдельных программ и прикладных пакетов Microsoft Office.

ЛАБОРАТОРНАЯ РАБОТА 1

Тема: использование встроенных функций MS Excel,
построение диаграмм

Необходимо составить ведомость расчета за месяц заработной платы сотрудников цеха по следующим условиям:

- табельный номер, ФИО, размер оклада задаются произвольно;
- принадлежность к профсоюзу определяет символ *;
- разряды работников с 1 по 6;
- премия насчитывается в соответствии с разрядом работника: 1..3 разряд – 20 % от оклада; 4 разряд – 35 % от оклада; 5, 6 разряд – 40 % от оклада.
- обязательный подоходный налог отчисления для всех работников – 13 %;
- если работник – член профсоюза, то ежемесячные отчисления составляют 1 %.

1. Откройте MS Excel и сохраните рабочую книгу под именем Лаб_1.xlsx.
2. В ячейки рабочего листа введите следующий текст (рис. 1).

The screenshot shows an Excel spreadsheet with the following content:

| Начисление заработной платы сотрудникам | | | | | | | | | |
|---|-----|------------------|--------|-------|--------|------------------|--------------------|----------|--|
| | | подоходный налог | | 13% | | | | | |
| | | проф.взнос | | 1% | | | | | |
| таб. номер | ФИО | член профсоюза | разряд | оклад | премия | подоходный налог | профсоюзные взносы | к выдаче | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| Общий фонд заработной платы | | | | | | | | | |

Рис. 1. Форма Листа1

3. Рассчитайте премию и профсоюзные взносы, используя логическую функцию:

=ЕСЛИ (логическое_выражение; значение_истина; значение_ложь).

4. Вычислите подоходный налог у каждого работника, итоговую заработную плату и общий фонд заработной платы, используя функцию СУММ(адрес_блока_ячеек).

5. Постройте гистограмму по итоговой заработной плате работников. Подпишите ряды и легенду графика.

ЛАБОРАТОРНАЯ РАБОТА 2

Тема: создание макросов

Макрос – это средство, с помощью которого описывается произвольная последовательность действий пользователя для ее последующего многократного выполнения. Прежде чем использовать макросы, необходимо продумать выполняемые действия, т.е. составить алгоритм.

Запись макроса включается командой Вид – Макросы – Запись макроса.

Диалоговое окно **Запись макроса** имеет вид, показанный на рис. 2.

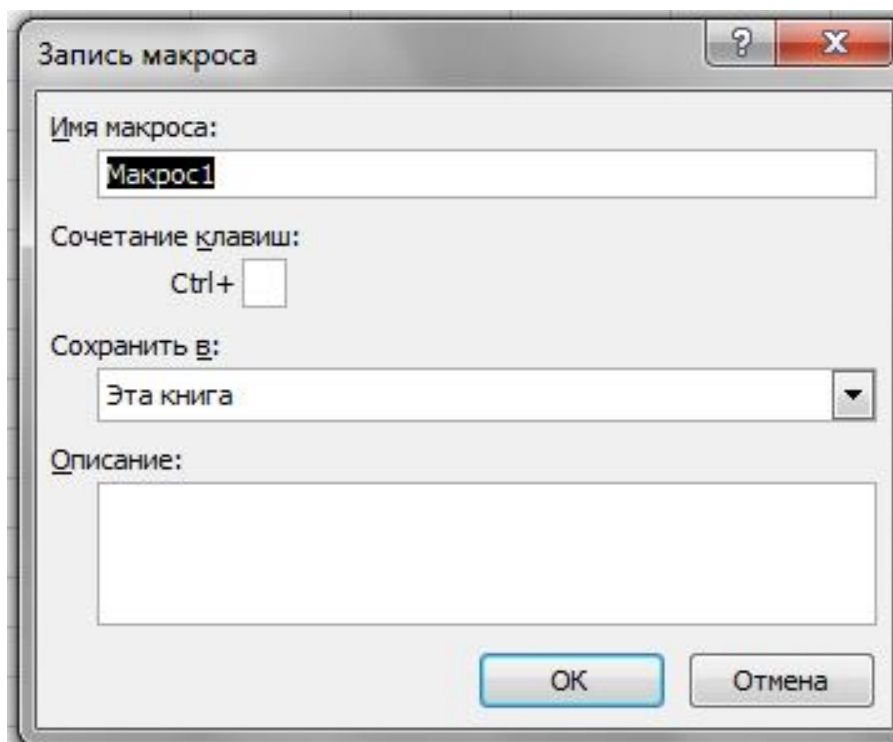


Рис. 2. Окно записи макроса

В поле **Имя макроса** указывается четко сформулированное имя по следующим правилам:

- должно начинаться с буквы, хотя внутри могут быть цифры и символ подчеркивания;
- не может содержать пробелы или знаки препинания (специальные символы);
- максимальная длина имени макроса – 80 символов.

По умолчанию VBA помещает в это поле значение **Макрос1**.

В поле **Сочетание клавиш** можно задать желаемую комбинацию клавиш, которая будет использоваться для быстрого вызова данного макроса.

В поле **Описание** записывается произвольный текст, т.е. комментарий о предназначении макроса.

Щелчок по кнопке **Ок** означает начало записи макроса.

После начала записи макроса необходимо последовательно выполнить все действия, которые впоследствии будут помещены в макрос и будут автоматически выполняться при каждом запуске данного макроса.

Когда требуемая последовательность действий будет выполнена, а значит, создаваемый макрос будет полностью записан, необходимо выбрать команду **Остановить запись** в меню **Вид – Макросы**.

Для редактирования текста макроса необходимо выбрать его имя в окне **Макросы** и нажать кнопку **Изменить** (рис. 3).

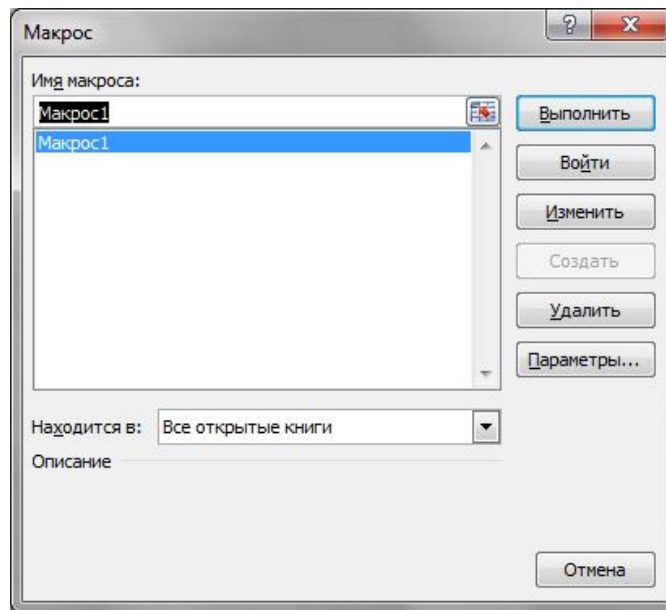


Рис. 3. Окно работы с макросами

При сохранении документа, содержащего макрос, в типе файла **Окна сохранения** необходимо указать **Книга Excel с поддержкой макросов**.

Кнопка запуска макроса позволяет организовать более быстрый доступ к его вызову. Для этого:

- на **Рабочий лист** добавить произвольную фигуру из меню **Вставка**;
- в контекстном меню фигуры (рис. 4) выбрать **Назначить макрос...** и из предложенного списка выбрать нужный макрос.

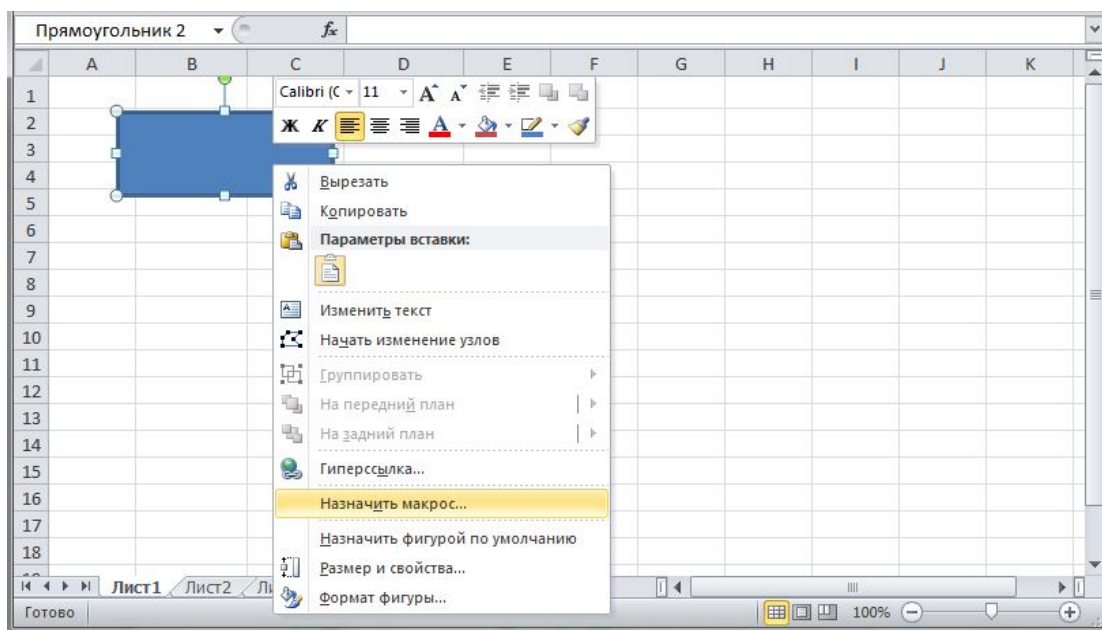


Рис. 4. Привязка макроса к кнопке

На кнопке вызова можно помещать любой поясняющий текст.

Задания к лабораторной работе 2

Пример 1. Создать макрос, который помещает в первую строку рабочего листа, начиная с ячейки В1, названия зимних месяцев. Завершается запись макроса в момент установления активной ячейки В3.

1. Создать Рабочую книгу Лаб_2.
2. Переименовать Лист1 в Месяцы.
3. Вызвать команду Запись макроса. В окне Запись макроса указать имя – ЗимниеМесяцы, в поле Сочетание клавиш введите символ М.
4. Щелкнуть по кнопке Ок для начала записи макроса.
5. Активируйте ячейку В1 и введите названия зимних месяцев, как на рис. 5. Залейте содержимое ячейки В1 желтым цветом, ячейки С1 – голубым, D1 – розовым. После перехода на ячейку В3 остановите запись макроса. Теперь созданный макрос готов к выполнению.

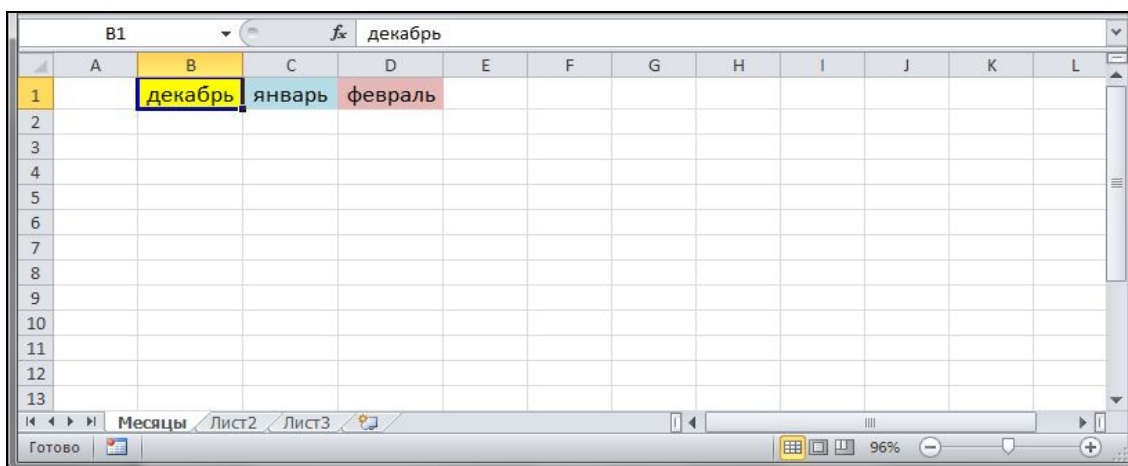


Рис. 5. Макрос ЗимниеМесяцы

Для проверки работы удалите данные и заливку с Листа1. В окне Макросы выберите из списка макрос ЗимниеМесяцы и нажмите кнопку Выполнить. Второй способ – нажмите сочетание клавиш Ctrl+M.

Пример 2. Рассчитать статистику продаж путевок во втором и третьем кварталах в санатории области (не менее пяти).

1. Переименовать Лист2 Рабочей книги Лаб_2 в Расчеты.
2. Создать таблицу, начиная с ячейки В2, с заголовками: Санаторий, Апрель, Май, Июнь, Июль, Август, Сентябрь, Среднее значение за I квартал, Среднее значение за II квартал.
3. Заполнить произвольными значениями столбцы В, С, D, E, F, G, H.
4. Добавить макрос Формула1, который рассчитывает среднее значение посещений каждого санатория за I квартал, и макрос Формула2 - среднее по посещениям за II квартал. Использовать функцию СРЗНАЧ(адрес_блока_ячеек).

5. Построить круговую диаграмму для средних значений по каждому кварталу.

Дополнительные задания

1. Откройте рабочую книгу Лаб_1.xlsx.
2. Переименуйте Лист2 в Макросы.
3. Создайте макросы:
 - **Заголовок**, автоматизирующий ввод в ячейку В1 текста **Начисление заработной платы сотрудникам** (см. Лист1 рабочей книги Лаб_1);
 - **Таблица**, определяющий границы обеих таблиц и заполнение заголовков столбцов (см. рис. 1);
 - **ВводДанных**, заполняющий столбцы А, В, С, D, Е таблицы произвольными значениями.
 - **Формулы**, вычисляющий расчетные значения.
4. Добавьте управляющие кнопки и назначьте каждой соответствующий макрос (рис. 6).

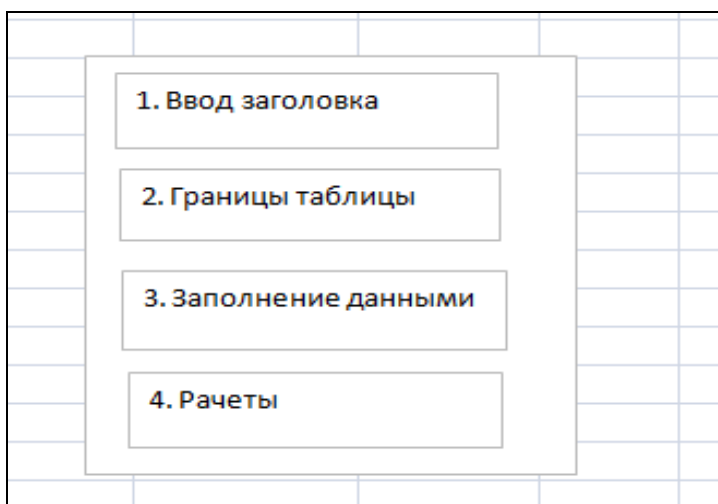


Рис. 6. Управляющие кнопки

5. Проверьте работу приложения.

ЛАБОРАТОРНАЯ РАБОТА 3

Тема: редактор VBA

Редактор Visual Basic of Application (VBA) позволяет записывать, сохранять и модифицировать программные модули, выполнять и отлаживать процедуры.

В редактор VBA можно перейти кнопкой **Изменить** окна **Запись макроса**, предварительно выбрав имя макроса (рис. 7).

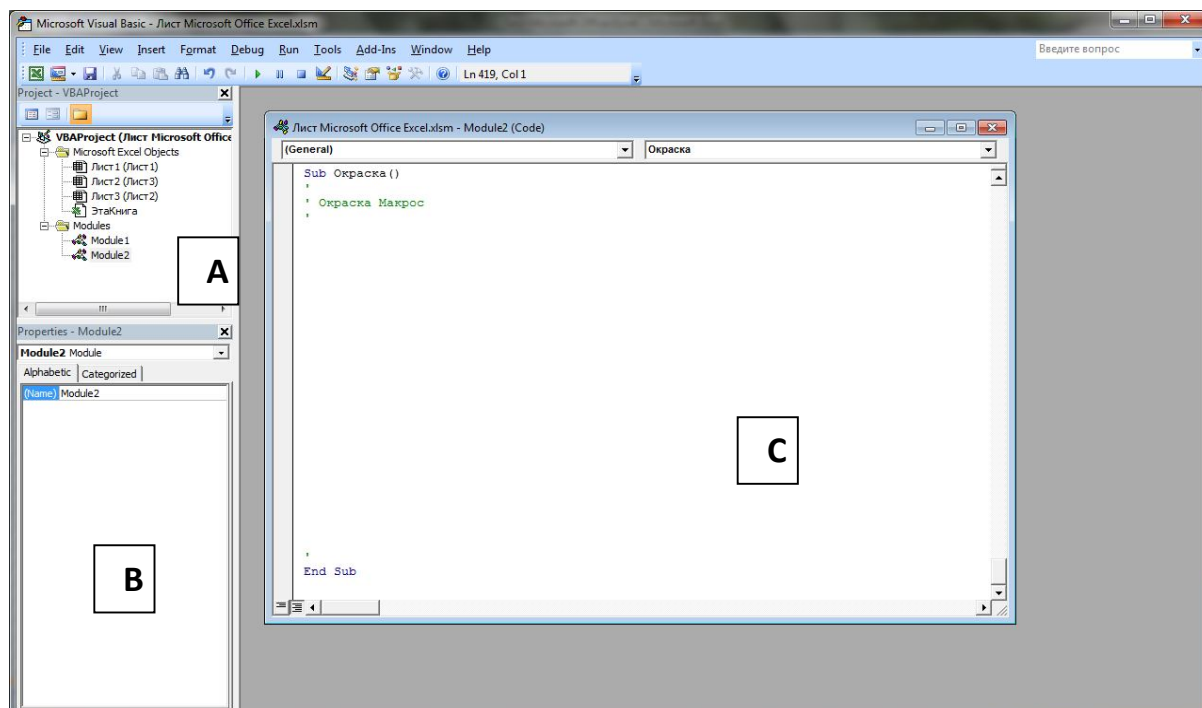


Рис. 7. Окно редактора VBA

Основные окна редактора VBA:

A - окно проекта (**Project Explorer**), предоставляющее быстрый доступ к окнам программного кода и пользовательским формам;

B - окно свойств (**Properties Windows**), позволяющее просмотреть и изменить значения свойств любого объекта любого типа (проекта, модуля, формы), который в данный момент активен;

C – окно программы (**Code**), позволяющее просматривать, редактировать и создавать исходный код на языке **VBA**.

Программа – это последовательность инструкций, которые компьютер выполняет одну за другой. Программа состоит из двух частей: объявление переменных и программный код. Любая **VBA**-программа обязательно содержит хотя бы одну процедуру.

Процедура – это отдельная единица программного кода **VBA**, содержащая один или несколько операторов.

Оператор – это наименьшая единица **VBA**-кода, предназначенная для определения переменной, установки параметров или выполнения какого-либо действия в программе.

Модуль – это именованная единица, состоящая из одной или нескольких процедур и раздела объявлений, в котором объявляются переменные, константы и пользовательские типы данных.

Проект включает в себя все модули, формы и связанные с приложением проекты, относящиеся к конкретному документу.

Переменная – это величина, которая хранится в памяти и значение которой может изменяться во время выполнения программы. Каждая переменная имеет свой тип, который определяется типом данных, сохраняемых в ней. В табл.1 приведены типы и диапазоны их значений.

Таблица 1

Типы данных VBA

| Тип | Наименование | Диапазон значений |
|--------------|---------------------------------------|--|
| Byte | байт | целое число от 0 до 255 |
| Boolean | логический | true (истина) или false (ложь) |
| Integer | целое | от -32768 до 32767 |
| Long | длинное целое | от -2 147 483 648 до 2 147 483 647 |
| Single | с плавающей точкой одинарной точности | от -3.402823E+38 до -1.401298E-45 (для отрицательных чисел) от 1.401298E-45 до 3.402823E+38 (для положительных чисел) |
| Double | с плавающей точкой двойной точности | от -1.79769313486232E+308 до -4.94065645841247E-324 (для отрицательных) от 4.94065645841247E-324 до 1.79769313486232E+308 (для положительных) |
| Currency | денежный | от -922 337 203 685 477.5808 до 922 337 203 685 477.5807 |
| Date | дата | от 01.01.100 до 31.12.9999 |
| String | строка | приблизительно 2 млрд символов |
| Variant | вариантный | любое число или любая строка |
| User-defined | любой | определяется в соответствии с заданным типом данных |

Правила объявления переменных:

- максимальная длина имени переменной не может превышать 255 символов;
- каждая переменная имеет уникальное имя;
- первым символом имени переменной должна быть буква, за которой могут следовать символы любой комбинации (числа, буквы, знак «_»);
- имя переменной не может содержать пробелов, точек, восклицательных знаков, а также символов @, #, \$, %, &;
- в качестве имени переменной нельзя использовать ключевые слова VBA (Dim, as, for, if и т.д.).

Для объявления переменных используется оператор **Dim**:

```
Dim A as Integer
Dim B as String
```

Каждая программа должна быть написана таким образом, чтобы в ней легко можно было бы находить места, в которые нужно внести изменения.

Комментарий – текст, предназначенный для прочтения человеком и описывающий выполняемые действия. В VBA комментарии записываются после символа апострофа. Текст комментария будет оформлен зеленым цветом.

```
' описание переменных
Dim A as Integer      ' A – количество деталей
Dim B as String       ' B – день недели
```

При написании программы очень удобно, когда записываемый оператор виден в окне программного кода целиком. Если создаваемый оператор оказался слишком длинным, можно воспользоваться символом подчеркивания (_), который обозначает конец строки и следующая строка присоединена к текущей с образованием единого оператора. Чтобы символ подчеркивания воспринимался как таковой, необходимо перед ним обязательно ставить пробел.

```
Set Пункт = Новое_меню.Controls.Add _
            (Type:=msoControlPopup)
```

Арифметические операторы позволяют выполнить любые арифметические действия в полном арифметическом соответствии правилам арифметики (табл. 2).

Таблица 2

Арифметические операторы

| Оператор | Описание |
|----------|------------------------------|
| + | сложение |
| - | вычитание |
| * | умножение |
| / | деление |
| \ | деление без остатка |
| ^ | возведение в степень |
| mod | остаток от деления по модулю |
| sqr | корень |

Логические операторы используются в VBA для манипулирования логическими значениями TRUE (в числовом значении это 1) и FALSE (в числовом значении это 0). Результатом логических операций является значение типа Boolean.

Логические операторы в VBA

| Оператор | Описание |
|----------|-----------------|
| And | и |
| Or | или |
| Not | нет |
| <> | неравно |
| Xor | исключение |
| Eqv | эквивалентность |
| Imp | импликация |

Например,

```
(a>3) and (b<0)
(c<0) or (s>=5)
```

Строковые операторы. Для слияния двух строк в одну используется оператор конкатенации (знак “ & ”). Результат этой операции всегда имеет тип String.

Например,

```
FIO = “Иванов” & “ ” & “Иван ” & “ ” & “Иванович”
```

Результат: в переменную FIO будет помещена строка “Иванов Иван Иванович”.

Задания к лабораторной работе 3

1. Откройте рабочую книгу Лаб_1.xlsx.
2. Создайте «пустой» макрос с именем Окраска.
3. Дополните модуль следующим текстом:

```
Sub Окраска()
Dim i As Integer
For i = 7 To 12
Cells(i, 9).Select
ActiveCell.EntireRow.Select
With Selection.Font
Select Case Cells(i, 9).Value

Case Is < 5000
.ColorIndex = 3

Case Is <= 10000
.ColorIndex = 10

Case Is <= 20000
.ColorIndex = 5

Case Else
.ColorIndex = 7
End Select
End With
Next i
End Sub
```

4. «Привяжите» макрос к кнопке и проверьте работу.
5. Подумайте о назначении программы и дополните модуль соответствующими комментариями.

Дополнительные задания

1. Добавьте рабочую книгу с именем Лаб_31.xlsx.
2. Создайте таблицу. В ячейку А6 введите №, В6 – ФИО, в ячейки С6:Н6 наименования дисциплин, в ячейку I6 – Средний балл. Необходимо заполнить таблицу успеваемости за сессию семи студентов и рассчитать средний балл.
3. Создайте макрос Цвет, по выполнению которого значения среднего балла диапазона [1;3] окрасятся желтым, [3,1;4] – зеленым; [4,1;5] – синим цветом
4. Проверьте работу.

ЛАБОРАТОРНАЯ РАБОТА 4

Тема: знакомство с диалоговыми функциями InputBox(), MsgBox().

Работа с ячейкой. Оператор присваивания. Линейный алгоритм

Встроенная функция **InputBox()** высвечивает диалоговое окно для ввода значения. Например, InputBox(“введите возраст”) имеет результат, представленный на рис. 8.

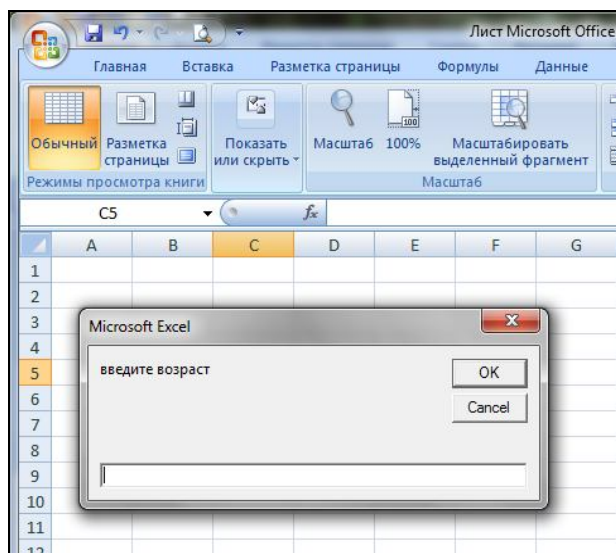


Рис. 8. Функция InputBox()

Встроенная функция **MsgBox()** используется для вывода сообщений. Текст сообщения является строкой символов.

Например, MsgBox («Наступил год Змеи») имеет результат, представленный на рис. 9.

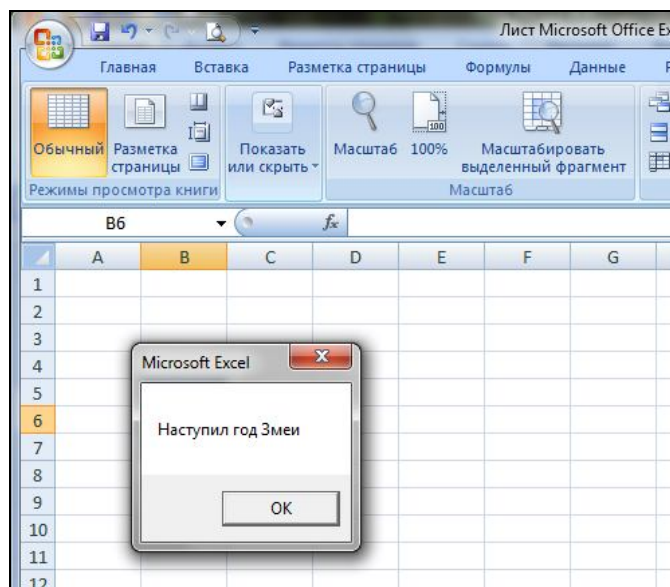


Рис. 9. Функция MsgBox()

Свойство **Cells()** возвращает единственную ячейку рабочего листа, которая находится на пересечении строки и столбца, задаваемых целыми числами в аргументе.

Cells (RowIndex, ColumnIndex)

Здесь

RowIndex – индекс строки;

ColumnIndex – индекс столбца.

Например,

Cells(1, 1) соответствует ячейке на пересечении первой строки и первого столбца), т.е. ячейке A1,

Cells(5, 7) соответствует пересечению пятой строки и седьмого столбца, т.е. ячейке G5,

Cells(11, 16) соответствует пересечению одиннадцатой строки и шестнадцатого столбца, т.е. ячейке P11.

Оператор присваивания. Процесс установки значения переменной называется присвоением. Оператор присваивания составляет основу всех вычислений.

Оператор присваивания вычисляет значение выражения справа и назначает его переменной, стоящей слева, при этом прежнее значение этой переменной заменяется на новое. В общем виде оператор присваивания имеет синтаксис:

переменная = выражение (или формула)

Например,

| |
|--|
| $A = B + C + \ln(B)$ $K = \text{Cells}(1, 3)$ |
|--|

Задания к лабораторной работе 4

Пример 1

1. Создайте рабочую книгу с именем Диалоговые функции.
2. Переименуйте Лист1 рабочей книги в InputBox.
3. Добавьте «пустой» макрос с именем ДиалогВвода и введите следующий текст:

```
Sub ДиалогВвода()  
Cells(1, 2) = InputBox("Введите свои ФИО")  
End Sub
```

4. Добавьте кнопку, привяжите ее к исполнению макроса ДиалогВвода и проверьте работу.

Пример 2

1. Переименуйте Лист2 рабочей книги в MsgBox.
2. Добавьте «пустой» макрос с именем ВыводСообщения и введите следующий текст:

```
Sub ВыводСообщения()  
MsgBox("Поздравляю, Вы освоили диалоговые функции VBA")  
End Sub
```

3. Создайте кнопку и «привяжите» ее к исполнению макроса Вывод Сообщения, проверьте работу.

Пример 3

1. Переименуйте Лист3 в Присваивание.
2. Создайте «пустой» макрос с именем ОператорПрисваивания со следующим программным кодом:

```
Sub ОператорПрисваивания()  
Cells(8, 2) = "Иванов"  
Cells(10, 3) = "Иван"  
Cells(12, 4) = "Иванович"  
Cells(14, 5) = "26 лет"  
End Sub
```

3. Добавьте кнопку и «привяжите» к ней макрос, проверьте работу.

Пример 4

Вычислить значение выражения

$$D = \frac{a^2 + b^2}{\sin(a) + c} - \sqrt{a \cdot b}$$

1. Переименуйте Лист4 в Линейный алгоритм.

2. В ячейку A1 введите текст a=, ячейку B1 залейте желтым цветом; в ячейку A2 введите текст b=, ячейку B2 залейте зеленым цветом; в ячейку A3 введите текст c=, ячейку B3 залейте голубым цветом; в ячейку A4 введите текст D=, ячейку B4 залейте розовым цветом (рис. 10).

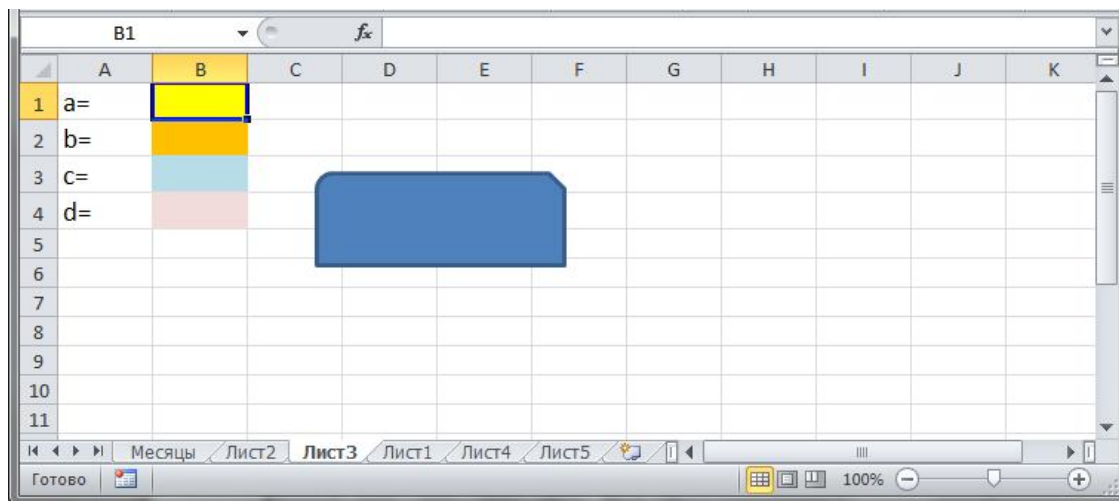


Рис. 10. Интерфейс приложения **ЛинейныйАлгоритм**

3. Создайте «пустой» макрос с именем **ЛинейныйАлгоритм** со следующим программным кодом:

```
Sub ЛинейныйАлгоритм()
Dim a, b, c, d As Double
a = Cells(1, 2)
b = Cells(2, 2)
c = Cells(3, 2)
D = (a * a + b * b) / (Sin(a) + c) - Sqr(a * b)
Cells(4, 2) = D
End Sub
```

4. «Привяжите» макрос к кнопке и проверьте работу.

Пример 5

Вычислить значение выражения

$$f = \frac{\sqrt{x^2 + a^2 + b^2}}{a - b} + \frac{1}{\sin(x) \cdot \cos(b)}$$

1. Переименуйте **Лист5** в **Линейный алгоритм2**.

2. В ячейку A1 введите текст a=, ячейку B1 залейте зеленым цветом; в ячейку A3 введите текст b=, ячейку B3 залейте розовым цветом; в ячейку A5 введите текст x=, ячейку B5 залейте сиреневым цветом; в ячейку A7 введите текст f=, ячейку B7 залейте желтым цветом.

3. Создайте «пустой» макрос с именем **ЛинейныйАлгоритм2** со следующим программным кодом:

```
Sub ЛинейныйАлгоритм2()
Dim a, b, x, f As Double
```

```
a = Cells(1, 2)
b = Cells(3, 2)
x = Cells(5, 2)
f = (Sqr(x * x + a * a + b * b) / (a - b)) + 1 / (Sin(x) * Cos(b))
Cells(7, 2) = f
End Sub
```

4. «Привяжите» макрос к кнопке и проверьте работу.

Дополнительные задания

Задание 1

1. Создайте рабочую книгу с именем Лаб_41.xlsx.
2. На Лист1 добавьте макрос ФИО, позволяющий через диалоговое окно InputBox() в ячейку B4 вывести Вашу фамилию, в ячейку C7 – имя, в ячейку D11 – отчество.
3. На Лист2 добавьте макрос ДатаРождения, при выполнении которого в окне MsgBox() появляется дата Вашего рождения.
4. Проверьте работу.

Задание 2

1. Создайте рабочую книгу с именем Лаб_42.xlsx.
2. Вычислите значение выражения согласно варианту.

ЛАБОРАТОРНАЯ РАБОТА 5

Тема: управляющие операторы VBA

Условный оператор IF организует выполнение блока операторов, если заданное условие выполняется и не делает ничего, если заданное условие не выполняется. В общем виде:

| | |
|------------------------|----------------------|
| If условие Then | |
| | <i>(Операторы 1)</i> |
| Else | |
| | <i>(Операторы 2)</i> |
| End If | |

Условие – это условное выражение, которое VBA должен вычислить. Если вычисленное значение **ИСТИННО (True)**, выполняется блок операторов, следующий за словом **Then**, если значение **ЛОЖНО (False)**, выполняется блок операторов, следующий за словом **Else**.

End If – завершающий оператор.

Если оператор **If** простой, то **Else** не записывается.

Например, вычислить значение переменной *c*:

```

If a > b then
    c = a * b
else
    c = a + b
End If
    
```

1) $a = 3, b = 2$ подставим значения в If:

If $3 > 2$ Then $c = 3 \cdot 2$

Else $c = 3 + 2$

End If

Поскольку условие $3 > 2$ истинно (True), ответом будет блок операторов, стоящий после Then, $c = 3 \cdot 2 = 6$.

2) $a = 4, b = 6$ подставим значения в If:

If $4 > 6$ Then $c = 4 \cdot 6$

Else $c = 4 + 6$

End If

Поскольку условие $4 > 6$ ложно (False), ответом будет блок операторов, стоящий после Else, $c = 4 + 6 = 10$.

Оператор Select ... Case используется в том случае, когда необходимо проверять одно и то же значение, сравнивая его с различными выражениями. Общий синтаксис оператора следующий:

```

Select Case значение
    Case условие1
        (операторы1)
    Case условие2
        (операторы2)
    .....
    Case условиеN
        (операторыN)
Case Else
        (операторы_else)
End Select
    
```

Сначала вычисляется значение, которое может быть любым числовым или строковым выражением, а затем полученный результат сравнивается с каждым из условий в операторах Case, начиная с первого условия1. Если результат вычисления выражения значение совпадет со значением выражения условие1, будут выполнены операторы блока (операторы1), после чего выполнение программы продолжится, начиная с оператора, следующего за оператором End Select.

Если результат вычисления выражения **значение** не совпадает ни с одним из значений выражений, начиная с выражения **условие1** и заканчивая выражением **условиеN**, то будет выполнен блок операторов **операторы_else**, следующий за ключевым словом **Case Else**. Оператор **Case Else** всегда находится в конце конструкции **Select Case** и не является обязательным, т.е. он и соответствующий ему блок операторов могут быть опущены.

В каждом операторе **Case** выражение **условие** можно представить следующим образом:

- любое выражение (строковое, числовое или логическое). В этом случае результат вычисления выражения **значение** должен полностью совпасть с результатом вычисления значения выражения **условие**;
- выражение, задающее диапазон значений. В этом случае указываются граничные значения диапазона, разделяемые ключевым словом **То**. Например, запись **10 То 20** означает, что проверяемое значение должно попасть в диапазон чисел от 10 до 20;
- конструкция с ключевым словом **Is**. Например, **Is > 50** означает, что сравнивается само значение с 50.

Задания к лабораторной работе 5

Пример 1

1. Создайте рабочую книгу с именем **УсловныеОператоры**.
2. Переименуйте **Лист1** в **Оператор_IF**.
3. Создайте макрос с именем **Опер_IF** и программным кодом:
Sub Опер_If()
Dim A, B, C As Integer
A = InputBox("Введите A")
B = InputBox("Введите B")
If A > B Then C = A + B
If A <= B Then C = A * B
Cells(3, 3)=C
End Sub
4. «Привяжите» макрос к кнопке и проверьте работу.

Пример 2

1. Переименуйте **Лист2** в **Оператор_Case**.
2. Создайте макрос с именем **Опер_Case** и программным кодом:
Sub Опер_Case()
Dim A as Single
A = InputBox("Введите рост в см")
Select Case A
Case Is < 150
MsgBox "Маленький рост"
Case 151 To 175
MsgBox "Средний рост"

```
Case Is > 175
MsgBox "Высокий рост"
End Select
End Sub
```

3. «Привяжите» макрос к кнопке и проверьте работу.

Пример 3

Вычислить значение переменной y :

$$y = \begin{cases} \sqrt{x+a}, x \geq 0, a \geq 0 \\ x^2 + a^2, x < 0 \\ \frac{1}{x+a}, x = 5, a < 0 \end{cases}$$

1. Переименуйте Лист3 в АлгоритмРазветвления.

2. В ячейку A1 введите текст $a=$, ячейку B1 залейте желтым цветом; в ячейку A3 введите текст $x=$, ячейку B3 залейте розовым цветом; в ячейку A5 введите $y=$, ячейку B5 залейте голубым цветом (рис. 11).

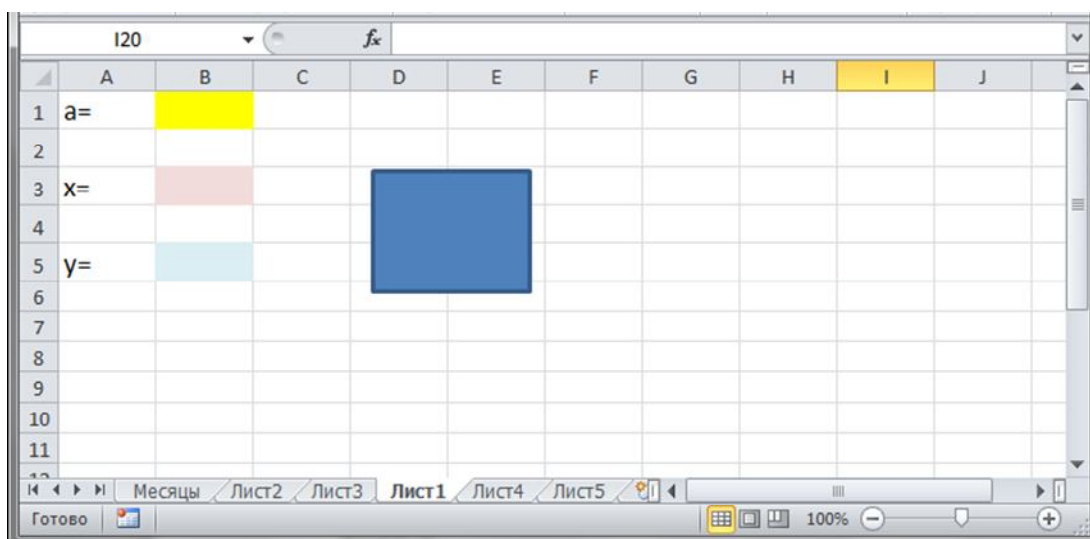


Рис. 11. Интерфейс приложения АлгоритмРазветвления

3. Создайте макрос **ВычислитьY** с программным кодом:

```
Sub ВычислитьY()
Dim a, x, y As Double
a = Cells(1, 2)
x = Cells(3, 2)
If (x >= 0) And (a >= 0) Then y = Sqr(x + a)
If (x < 0) Then y = x * x + a * a
If (x = 5) And (a < 0) Then y = 1 / (x + a)
Cells(5, 2) = y
End Sub
```

4. «Привяжите» макрос к кнопке и проверьте работу приложения.

Пример 4

Вычислить значение переменной y :

$$y = \begin{cases} \log(x), & x > 1 \\ \frac{1}{x+2}, & x = 12 \\ \sin(x), & x \leq 1 \end{cases}$$

1. Переименуйте Лист4 в АлгоритмРазветвления2.
2. В ячейку А1 введите текст Переменная x , ячейку В1 залейте голубым цветом; в ячейку А3 введите $y=$, ячейку В3 залейте зеленым цветом.
3. Создайте макрос ВычислитьY2 с программным кодом:

```
Sub ВычислитьY2()
Dim x, y As Double
x = Cells(1, 2)
If x > 1 Then y = Log(x)
If x = 12 Then y = 1 / (x + 2)
If x <= 1 Then y = Sin(x)
Cells(3, 2) = y
End Sub
```
4. «Привяжите» макрос к кнопке и проверьте работу приложения.

Дополнительные задания

Используя условный оператор, вычислите на Листе4 рабочей книги УсловныеОператоры значение переменной y согласно своему варианту.

ЛАБОРАТОРНАЯ РАБОТА 6

Тема: циклические алгоритмы

Циклом называется алгоритм, который обеспечивает повторение каких-либо действий.

Циклы бывают трех видов: с предусловием, с постусловием, с параметром.

Цикл с параметром используется, когда число повторений цикла известно до начала его работы. При создании такого цикла надо определить его переменную, которая служит **счетчиком** числа повторений цикла, и задать ее начальное и конечное значения. В качестве дополнительного параметра можно указывать величину изменения счетчика – **шаг (Step)**, не является обязательным параметром. Если шаг равен 1, то его можно не писать.

В общем виде:

```
For счетчик=нач_значение TO кон_значение STEP шаг изменения
операторы_цикла
Next счетчик
```

При программировании иногда сложно сказать заранее, сколько повторяющихся действий необходимо выполнить. Чтобы добиться заданной точности, используются циклы, управляемые условием. Существует два основных типа условных циклов:

1) **цикл с предусловием Do While...Loop** выполняется до тех пор, пока условие истинно; операторы цикла могут не выполняться ни разу, если его условие окажется ложным при входе в цикл.

В общем виде:

```

счетчик=нач_значение
Do While условие
операторы_цикла
Loop
    
```

2) **цикл с постусловием Do...Loop Until** указывает, что цикл выполняется пока условие ложно; операторы цикла всегда будут выполнены, по крайней мере, один раз.

В общем виде:

```

счетчик=нач_значение
Do
операторы_цикла
Loop Until условие
    
```

Задания к лабораторной работе 6

Пример 1

Дан одномерный числовой массив B2:B14. Найти сумму элементов массива, используя все три вида циклов.

1. Создать рабочую книгу Циклы.xlsx.
2. Выделить желтым цветом массив B2:B14; в ячейку C1 ввести текст Сумма_For, ячейку D1 залить розовым цветом; в ячейку C2 ввести текст Сумма_While, ячейку D2 залить голубым цветом; в ячейку C3 ввести текст Сумма_Until, ячейку D3 залить зеленым цветом (рис. 12).

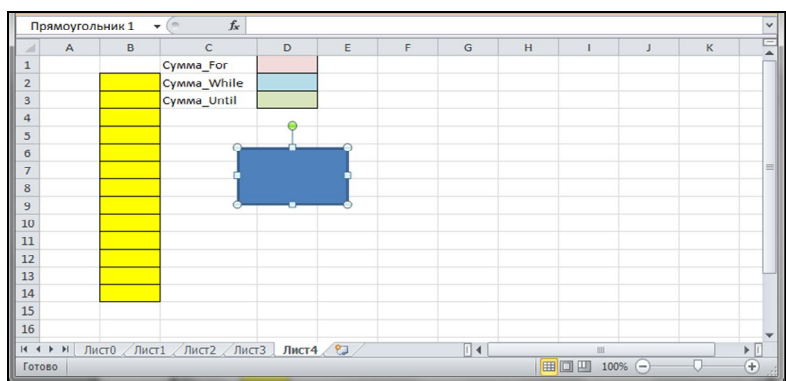


Рис. 12. Интерфейс приложения Циклы

3. Написать программный код для макроса **ОператорЦикла**:

```
Sub ОператорЦикла ()
Dim i, Sum1, Sum2, Sum3 As Double
' цикл For
Sum1 = 0
For i = 2 To 14
Sum1 = Sum1 + Cells(i, 2)
Next i
Cells(1, 4) = Sum1
' цикл While
Sum2 = 0
i = 2
Do While i < 15
Sum2 = Sum2 + Cells(i, 2)
i = i + 1
Loop
Cells(2, 4) = Sum2

' цикл Until
Sum3 = 0
i = 2
Do
Sum3 = Sum3 + Cells(i, 2)
i = i + 1
Loop Until i > 14
Cells(3, 4) = Sum3
End Sub
```

4. «Привязать» макрос к кнопке.
5. Проверить работу приложения.

Пример 2

Дан одномерный числовой массив В3:К3. Найти количество положительных элементов массива, ответ вывести в ячейку В5.

1. Перейти на новый лист рабочей книги **Циклы.xlsx**.
2. Выделить диапазон В3:К3, в ячейку А5 добавить текст **Количество положительных элементов**, ячейку В5 залить желтым цветом (рис. 13).

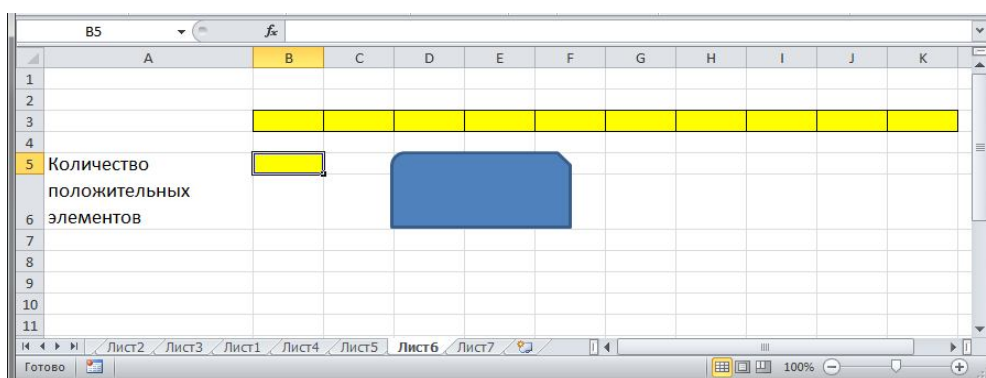


Рис. 13. Интерфейс приложения **Количество положительных элементов**

3. Добавить макрос **КоличествоПоложительных** с программным кодом:

```
Sub КоличествоПоложительных ()  
Dim j, kol As Double  
kol = 0  
For j = 2 To 11  
If Cells(3, j) > 0 Then kol = kol + 1  
Next j  
Cells(5, 2) = kol  
End Sub
```

4. «Привязать» макрос к кнопке и проверить работу.

Пример 3

Дан одномерный числовой массив G4:G17. Найти максимальный элемент массива, ответ вывести в ячейку B4.

1. Перейти на новый лист рабочей книги Циклы.xlsx.

2. Выделить диапазон G4:G17, в ячейку A4 добавить текст **Максимальный элемент**, ячейку B4 залить голубым цветом.

3. Добавить макрос **МаксимальныйЭлемент** с программным кодом:

```
Sub МаксимальныйЭлемент()  
Dim i, max As Double  
max = Cells(4, 7)  
For i = 4 To 17  
If Cells(i, 7) > max Then max = Cells(i, 7)  
Next i  
Cells(4, 2) = max  
End Sub
```

4. «Привязать» макрос к кнопке и проверить работу приложения.

Пример 4

Дан одномерный числовой массив D2:D21. Найти сумму нечетных отрицательных элементов. Ответ вывести в ячейке F5.

1. Перейти на новый лист рабочей книги Циклы.xlsx.

2. Выделить диапазон D2:D21, в ячейку E5 добавить текст **Сумма нечетных отрицательных**, ячейку F5 залить зеленым цветом.

3. Добавить макрос **СуммаНечетныхОтрицательных** с программным кодом:

```
Sub СуммаНечетныхОтрицательных()  
Dim i, Sum As Double  
Sum = 0  
For i = 2 To 21  
If (Cells(i, 4) Mod 2 <> 0) And (Cells(i, 4) < 0) Then Sum = Sum + Cells(i, 4)  
Next i  
Cells(5, 6) = Sum  
End Sub
```

4. «Привязать» макрос к кнопке и проверить работу.

Пример 5

Дан одномерный числовой массив C7:C19. Вывести в диапазон D7:D19 элементы массива с четным порядковым номером.

1. Перейти на новый лист рабочей книги Циклы.xlsx.
2. Выделить диапазон C7:C19 розовым цветом, а диапазон D7:D19 – голубым.
3. Добавить макрос ЭлементыЧетныйНомер с программным кодом:
Sub ЭлементыЧетныйНомер()
Dim i As Double
For i = 7 To 19
If i Mod 2 = 0 Then Cells(i, 4) = Cells(i, 3)
Next i
End Sub
4. «Привязать» макрос к кнопке и проверить работу.

Пример 6

Дан одномерный числовой массив A10:H10. Найти сумму и количество элементов массива, находящихся в интервале [-3;3]. Ответ вывести в ячейки B2 и B3 соответственно.

1. Перейти на новый лист рабочей книги Циклы.xlsx.
2. Выделить диапазон A10:H10, в ячейку A2 добавить текст Сумма, ячейку B2 залить желтым цветом; в ячейку A3 – Количество, ячейку B3 залить голубым цветом.
3. Добавить макрос СуммаКоличествоИнтервал с программным кодом:
Sub СуммаКоличествоИнтервал()
Dim j, Sum, Kol As Double
Sum = 0
For j = 1 To 8
If (Cells(10, j) >= -3) And (Cells(10, j) <= 3) Then Sum = Sum + Cells(10, j)
Next j
Cells(2, 2) = Sum
Kol = 0
For j = 1 To 8
If (Cells(10, j) >= -3) And (Cells(10, j) <= 3) Then Kol = Kol + 1
Next j
Cells(3, 2) = Kol
End Sub
4. «Привязать» макрос к кнопке и проверить работу.

Дополнительные задания

Используя оператор цикла, на Листе7 рабочей книги Циклы.xlsx выполните индивидуальное задание согласно своему варианту.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Белоусова С.Н. Основные принципы и концепции программирования на языке VBA в Excel: Учебное пособие / С.Н. Белоусова, И.А. Бессонова – М.: Интернет-Университет Информационных технологий: БИНОМ. Лаборатория знаний, 2010. – 200 с.
2. Слепцова Л.Д. Программирование на VBA в Microsoft Excel 2010. – М.: ООО «ИД Вильямс», 2010. – 432 с.