

Создание макросов на языке VBA

Практикум по решению задач на ЭВМ

Г.В.Гибатулина

Киров - 2013

*

Понятие макроса

Макрос (или макрокоманда) – это набор действий, которые можно выполнить с помощью одной команды, имеющей имя

В Microsoft Office 95 и 97 макросы записывались на языке WordBasic, а начиная с MS Office 2000 – на языке **VBA** (Visual Basic for Application).

Назначение макросов

- ускорение часто выполняемых операций редактирования или форматирования (стили)
- объединение нескольких команд в одну
- упрощение доступа к параметрам в диалоговых окнах
- автоматизация обработки сложных последовательных действий в задачах

Операции с макросами

- Для быстрого запуска макроса можно создать для него кнопку, присвоить «горячую» клавишу или поместить макрокоманду в меню.
- Созданные макросы можно редактировать, сохранять, удалять, многократно запускать (**Сервис – макрос – макросы**).

Виды макросов

Командные – состоят из операторов, эквивалентных тем или иным командам меню или параметрам диалоговых окон. В результате выполнения макроса вносятся изменения либо в обрабатываемый документ, либо в среду приложения.

Пользовательские функции – аналогичны функциям Excel, т.е. используют значения передаваемых им аргументов, производят вычисления, возвращают результат в точку вызова, но не изменяют среду приложения.

Макрофункции – сочетание командных макросов и пользовательских функций.

Уровни безопасности по защите от макровирусов в пакете MS Office

Очень высокий – разрешаются запуск только макросов, установленных в надёжных расположениях.

Высокий – запускаются макросы, имеющие цифровую подпись. Если подписи нет, то макрос отключается. Электронный сертификат – специальное вложение, которое добавляется в макрос и служит для его защиты.

Средний – при открытии документа выдается сообщение о содержании макросов в данном документе. Пользователь сам выбирает, разрешить или нет использование макросов.

Низкий (не рекомендуется!) – все средства защиты от макровирусов отключаются.

Установка уровней защиты: **Сервис – Макрос - Безопасность**

Способы создания макросов

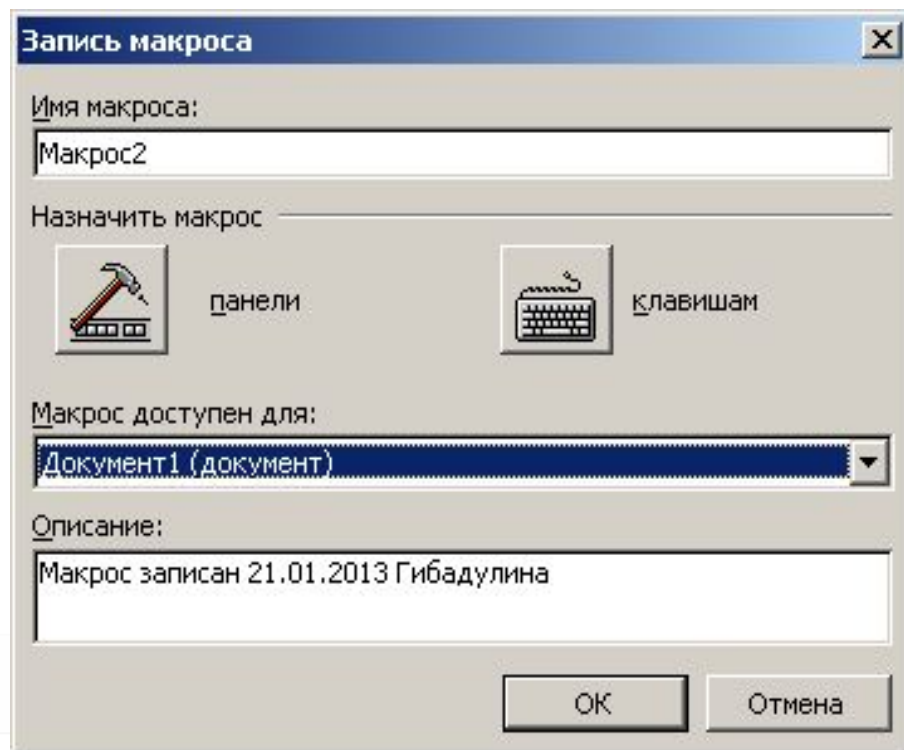
1. С помощью средств автоматической записи (не требует знания программирования)
2. Программирование на языке VBA

В приведённых примерах рассматривается MS Office 2003

A decorative grid pattern at the bottom of the slide, consisting of a 10x10 grid of squares. The squares are colored in a repeating pattern of light gray, light green, and light yellow.

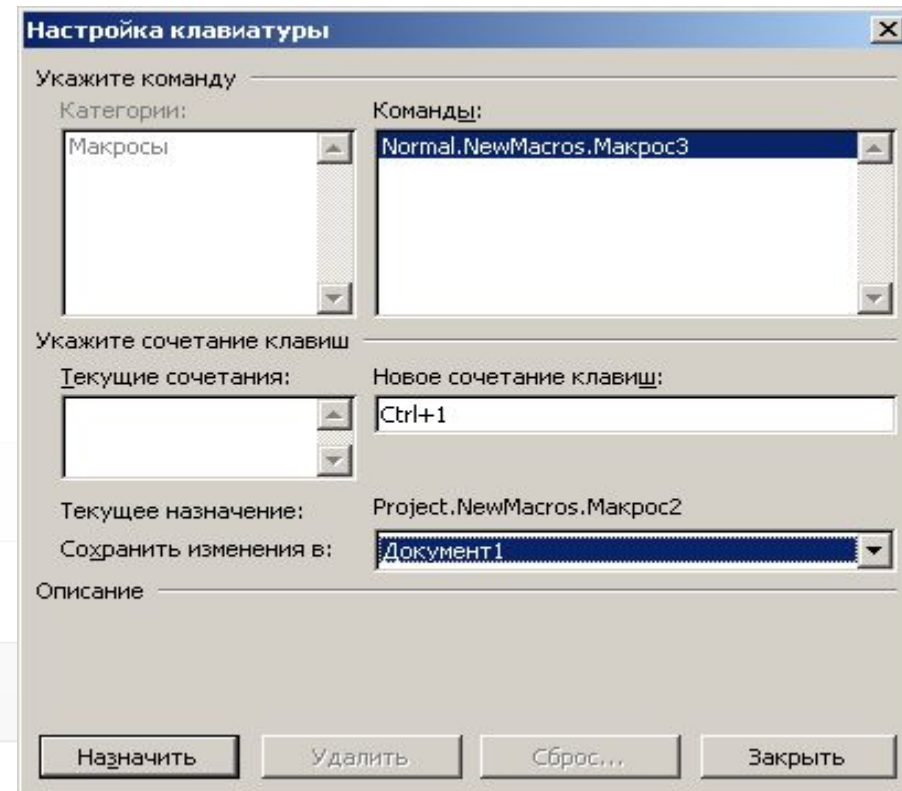
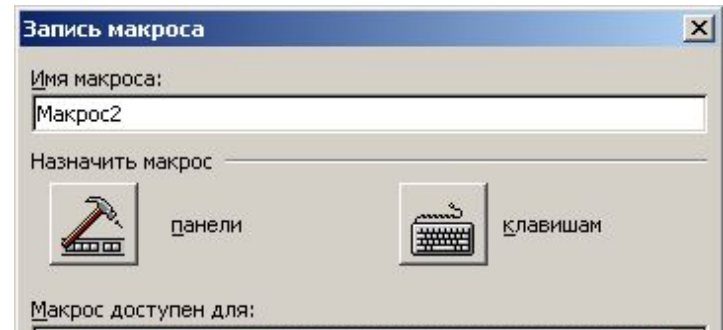
Автоматическое создание макроса

1. Выполнить команду **Сервис – Макрос - Начать запись**.
2. Ввести имя в поле «**Имя макроса**».
3. В списке «**Макрос доступен для**» выбрать шаблон или документ, в котором будет храниться макрос.
4. Ввести описание макроса в поле «**Описание**».
5. Выбрать **назначение макроса** либо Панели инструментов, либо Клавишам. Можно пропустить этот пункт, нажать кнопку ОК и перейти к записи макроса.



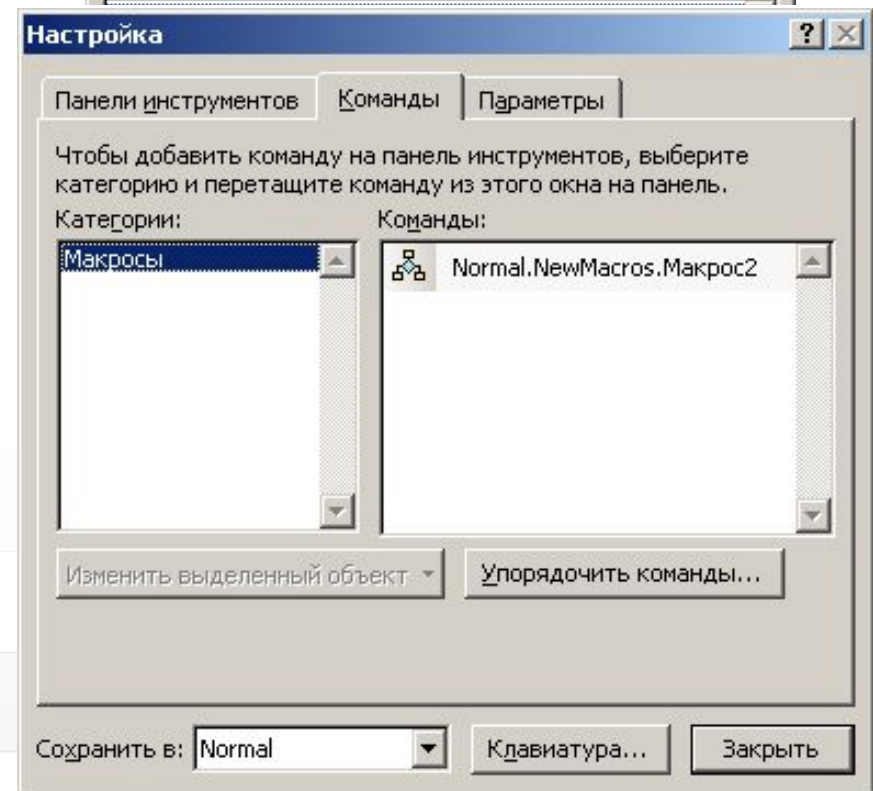
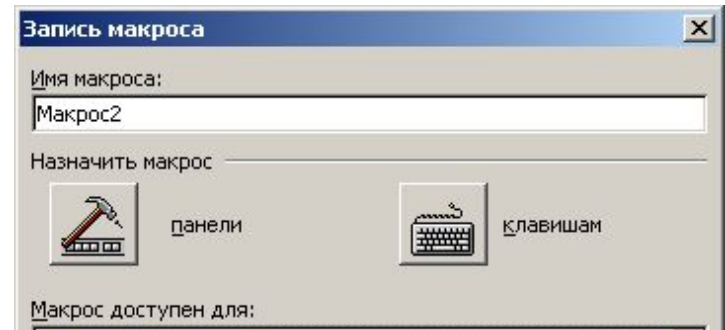
Назначение макроса сочетанию клавиш

1. В поле "Назначить макрос" нажмите кнопку "Клавишам".
2. Выберите записываемый макрос в списке "Команды", введите сочетание клавиш в поле "Новое сочетание клавиш".
3. В поле "Сохранить изменения в" выберите документ или шаблон, в котором будет доступен макрос.
4. Нажмите кнопку "Назначить" и затем "Заккрыть", чтобы начать запись макроса.



Назначение макроса панели инструментов

1. В поле "Назначить макрос" нажмите кнопку "Панели".
2. Выберите записываемый макрос в списке "**Команды**" и перетащите его на панель инструментов или в меню.
3. В поле "**Сохранить изменения в**" выберите документ или шаблон, в котором будет доступен макрос.
4. Нажмите кнопку "**Назначить**" и затем "**Заккрыть**", чтобы начать запись макроса.



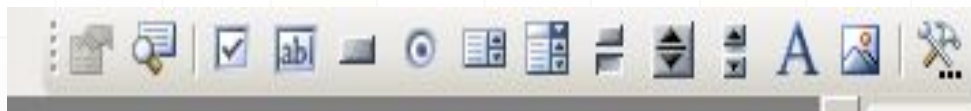
Создание макросов в Power Point

Задание

Создать проект «Проверь себя» для проверки навыков устного счёта и знания правил грамматики русского языка

Создание макросов в Power Point

1. Создайте слайд с заголовком «Математика».
2. Активизируйте панель инструментов "Вид - панели инструментов - элементы управления".
3. Разместите на слайде 4 командные кнопки (CommandButton1, 2, 3, 4).



Кнопка

Установка СВОЙСТВ КНОПОК

В контекстном меню кнопки выбрать Properties (свойства)

Properties

CommandBul CommandButt

Alphabetic Categorized

(Name)	CommandButto
Accelerator	
AutoSize	False
BackColor	<input type="checkbox"/> &#H8000000
BackStyle	1 - fmBackStyle
Caption	Деление
Enabled	True
Font	Arial
ForeColor	<input checked="" type="checkbox"/> &#H0000400
Height	48
Left	426
Locked	False
MouseIcon	(None)
MousePointer	0 - fmMousePoi
Picture	(None)
PicturePosition	7 - fmPicturePo
TakeFocusOnClick	True
Top	324
Visible	True
Width	150
WordWrap	False

Свойства

Математика

Проверка навыков устного счёта

Сложение

Умножение

Вычитание

CommandButton4

Выход

Создание пользовательской формы

UserForm1

Сложение
от 1 до

Введите ответ

Label2

Label15

Комментарий

TextBox1

Label12

Далее

Результат

Меню

Количество примеров

Количество верных ответов

Label4

CommandButton1

Label6

Label13

Программный код загрузки формы

```
Private Sub UserForm_Activate()    `активация формы
    n = 0    `количество верных ответов
    k = 1    `счётчик примеров
    z = Val(InputBox("Введите максимальную границу
для чисел от 10 до 1000"))
    Label2.Caption = Label2.Caption & Str(z)
    Randomize Timer
    a = Int(Rnd * 10)    `создание случайных чисел для
первого примера
    b = Int(Rnd * 10)
    s = a + b
    Label4.Caption = Str(a)    `вывод чисел в метки
    Label6.Caption = Str(b)
End Sub
```

Программный код кнопки «Далее»

```
Private Sub CommandButton1_Click()  
    If Val(s) = Val(TextBox1) Then    `проверка ответа  
        n = n + 1    `количество верных ответов  
        Label15.Caption = «Верно!»  
    Else  
        Label15.Caption = «Неверно!»  
    End If  
    k = k + 1    ` подсчёт количества примеров  
    Label12.Caption = ""    `очистка меток  
    Label13.Caption = ""  
    TextBox1 = ""    `очистка текстового поля для ответа  
    Randomize Time  
    a = Int(Rnd * z) + 1    `генерация случайных чисел  
    b = Int(Rnd * z) + 1  
    s = a + b    `сумма  
    Label4.Caption = Str(a)    `вывод чисел  
    Label6.Caption = Str(b)  
End Sub
```


Программный код кнопок «Результат» и «Назад»

```
Private Sub CommandButton2_Click()  
    ' Результат  
    Label12.Caption = Str(k)  
    Label13.Caption = Str(n)  
End Sub
```

```
Private Sub CommandButton3_Click()  
    ' Назад  
    End  
End Sub
```

Форма для проверки грамматики

UserForm5

Правописание "жи" и "ши"

Вставьте пропущенные буквы

е гули


кор вот

шки ж раф

мор ш рина

Количество верных ответов:

Проверка Назад Далее



Label1

Label2

TextBox1

Image1

TextBox8

Label15

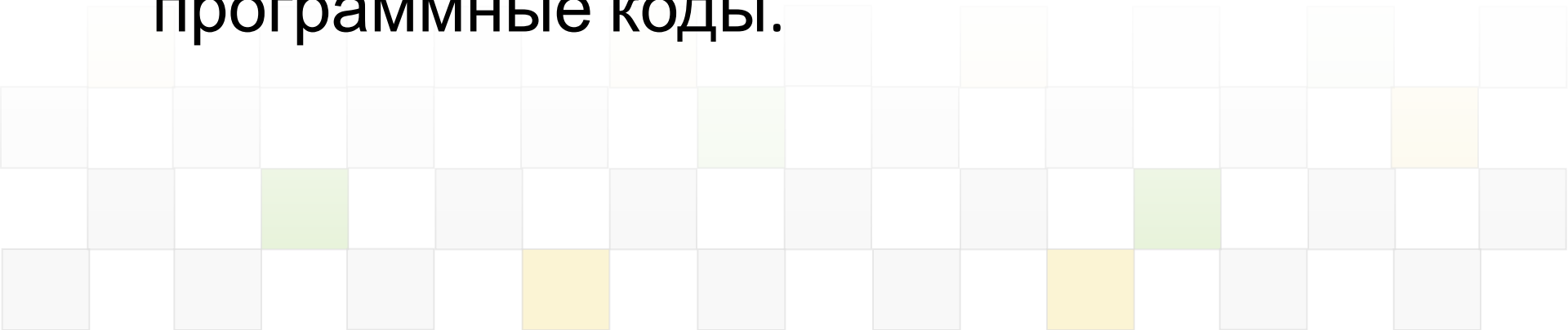
Label14

CommandButton1

Программный код кнопки «Проверка»

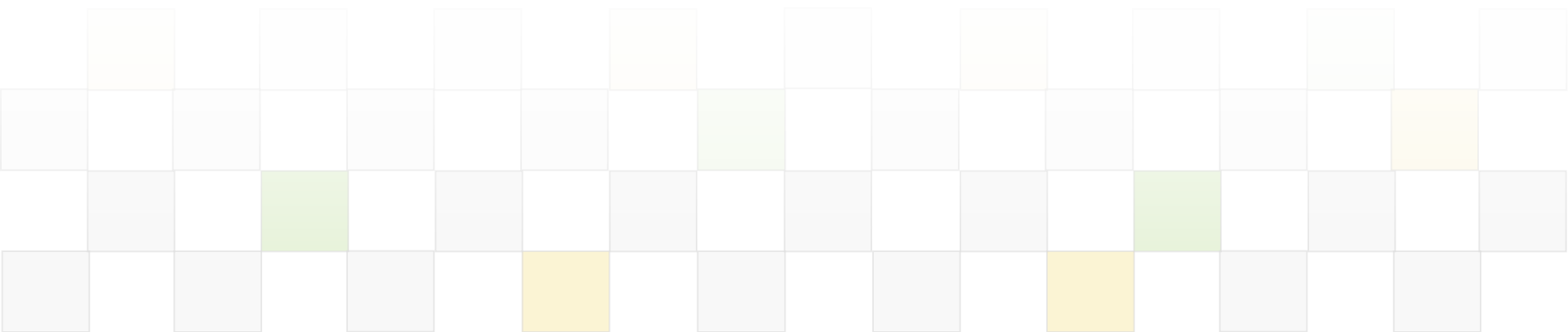
```
Private Sub CommandButton1_Click()  
    'Проверка  
    k = 0    `количество верных ответов  
    If TextBox1 = "жи" Then  
        k = k + 1: TextBox1.ForeColor = vbGreen  
        `выделение зелёным цветом верных ответов  
    Else  
        TextBox1.ForeColor = vbRed `выделение  
        красным цветом ошибок  
    End If  
    Label14.Caption = Str(k)  
    Label15.Caption = "Ошибки выделены красным  
цветом"  
End Sub
```

Задания для самостоятельной работы

1. Придумайте задания для проверки знаний по информатике.
 2. Подготовьте ребусы на любые темы.
 3. Создайте соответствующие слайды «Информатика» и «Ребусы», пользовательские формы и программные коды.
- 
- A decorative grid pattern at the bottom of the slide, consisting of a 10x5 grid of squares. The squares are colored in a repeating pattern of light green, light yellow, and light grey.

Пример проекта с макросами

Открыть проект «Проверь себя»



КОГОБУ СПО

«Кировский педагогический колледж»

Кировская обл., г. Киров,
ул. Свободы, 124
kpedk@mail.ru
<http://kp-kollege.ru>