МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Интеллектуальные информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №4

Тема: «СОЗДАНИЕ МАКРОСОВ WORD С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА VISUAL BASIC FOR APPLICATION»

Выполнил:

Студент 2 курса

Группы АС-59

Строганов Д.С.

Проверила:

Дряпко

Брест 2021

**Excele**

**Вариант 12**

Задание 1. Нахождения такой суммы S=1+2+3+…, для которой |S-M|  минимально. M –заданное число;

Public Function ïåðâîå(m)

Dim s, i, x As Double

s = 0

i = 1

x = m

Do

s = s + i

i = i + 1

x = Abs(s - m)

Loop Until Abs(s + i - m) >= x

ïåðâîå = s

End Function

**Вариант 4**

Задание 2.Решение линейного уравнения вида аx=c, где a и c - заданные

коэффициенты, в том числе и нулевые

Код макроса:

Public Function вторая(a As Variant, c As Variant) As Variant

Dim Error As String

If a = 0 And c <> 0 Then

Error = "Undefined"

MsgBox Error

вторая = "-"

Else

If a = 0 And c = 0 Then

Error = "Any Rational number"

MsgBox Error

вторая= "-"

Else

вторая = c / a

End If

End If

End Function

**Вариант 13**

Задание 3. На отрезке [n, m] найти все автоморфные числа.

Public Function третья(n As Double, m As Double) As String

'На отрезке [n, m] найти все автоморфные числа

Dim i As Long

Dim s As String

For i = n To m

If третья1(i) Then s = s + Format(i) + " "

Next i

третья = s

End Function

Public Function третья1(n As Long) As Variant

'Функция выясняет, является ли заданное число Автоморфным

'Число автоморфное, если после возведения в квадрат оно совпадает

'с младшими разрядами числа (5^2=25, 25^2=625)

If n ^ 2 Mod 10 ^ количество(n) = n Then

третья1 = True

Else

третья1 = False

End If

End Function

Public Function количество(ByVal n As Long) As Variant

'Функция вычисляет количество цифр числа

количество = 0

While n > 0

n = n \ 10

количество = количество + 1

Wend

End Function

**Вариант 1**

Задание 4. Нахождения суммы положительных (отрицательных) элементов массива;

Sub laba4 ()

Dim a() As Integer

Dim n, i As Integer

Dim summa, proiz As Integer

Do

n = Val(InputBox("vvedi kol-vo el-tov"))

Loop Until n > 0

ReDim a(n)

For i = 1 To n

a(i) = Val(InputBox("vvedi " + Str(i) + \_

"-й el-t massiva"))

Next

For i = 0 To n

Next

summa = 0

proiz = 1

For i = 0 To n

summa = summa + a(i)

Next

MsgBox (summa)

End Sub

**Вариант 11**

Задание 5. Найти сумму элементов каждой строки матрицы;

Sub laba5()

Dim A() As Double, L As Long, Sum As Double

ReDim A(1 To 4, 1 To 4)

A(1, 1) = -3: A(1, 2) = 3: A(1, 3) = 3: A(1, 4) = 2

A(2, 1) = 5: A(2, 2) = -6: A(2, 3) = 2: A(2, 4) = 4

A(3, 1) = -3: A(3, 2) = -1: A(3, 3) = -2: A(3, 4) = -6

A(4, 1) = 4: A(4, 2) = -5: A(4, 3) = 2: A(4, 4) = -3

For L = 1 To 4

Sum = A(L, 1) + A(L, 2) + A(L, 3) + A(L, 4)

MsgBox "Сумма элементов строки равна" & L & " = " & Sum

Next L

End Sub

**Вариант 6**

Задание 6. Даны строки S1, S2 и символ C. Перед (после) каждого вхождения символа C в строку S1 вставить строку S2.

Код макроса:

Sub Zadanie()

Dim S1 As String

Dim S2 As String

Dim C As String

S1 = InputBox("Stroka1?")

S2 = InputBox("Stroka2?")

C = InputBox("Simvol?")

S = Replace(S1, C, C + S2)

MsgBox (S)

End Sub

**Вариант 6**

Задание 7. В заданном тексте найти слова заданной длины **n**. Исходный текст задается в элементе управления ***поле,*** число **n** выбирать из элемента управления ***счетчик***,  найденные слова разместить в ***списке*** пользовательской формы.

Private Sub MultiPage1\_Change()

End Sub

Private Sub счетчик\_Change()

Заданиебокс2 =счетчик.Value

End Sub

Private Sub CommandButton8\_Click()

Dim s, st As String

Dim i, n, k As Double

Заданиебокс1.MultiLine = True

s = Заданиебокс1.Value

s = " " + s + " "

st = ""

k = 0

n = Val(заданиебокс2.Value)

With Заданиебокс1

заданиебокс1.Clear

For i = 2 To Len(s)

If Mid(s, i, 1) = " " Then

If Mid(s, i - 1, 1) <> " " Then

If k = n Then

.AddItem Mid(s, i - n, n)

End If

k = 0

End If

Else

k = k + 1

End If

Next i

End With

End Sub

Word

**Вариант 4**

**4.** Составить программу печати документа книжкой. Для этого надо выполнить настройки параметров страницы таким образом, чтобы после распечатки документа листы можно было посередине сшить. Перед началом печати пользователем  в диалоговом окне вводится желаемая ширина всех 4-х полей для одной стр., т.е. половины листа. Использовать вид элементов управления формы для ввода – поле увеличения-уменьшения (поле со счетчиком), предусмотреть значение полей по умолчанию.

Основа алгоритма печати следующая:

1. установить альбомную ориентацию страницы;
2. организовать 4 цикла печати. Для каждого цикла задаются свои размеры полей печати. В каждом цикле печатается 1/4 от общего количества страниц документа.

Например, в исходном файле 8 стр. Для распечатки его нужно 2 листа - по 2 стр. на лист с двух сторон. Длина листа формата А4 в альбомной ориентации - 297 мм, половина листа - 148 мм.

Если краевое поле каждой части стр. должно составлять 15 мм, а серединное - 30 мм, то ширина одной стр. текста - 148-30-15=103 мм.

1 цикл - печать страниц 1 и 3. Поля: левое-15 мм, правое - 297-103-15=179 мм (см. рис. 2)

Здесь размещается текст страниц 1 и 3

Рисунок 2 - Левое и правое поля для 1 цикла печати

2 цикл - печать 6 и 8 стр. на тех же сторонах двух листов. Поля: левое - 148+15=163 мм, правое -30 мм (см. рис.3).

Здесь размещается текст страниц 6 и 8

Рисунок 3 - Левое и правое поля для 2 цикла печати

Настройки 3 и 4 циклов аналогичны.

Если число страниц в документе не кратно 4, то в конец документа добавляются пустые страницы.

Код программы :

Sub лабаворд()

Dim a, b, c, d As Integer

ActiveDocument.Sections(1).PageSetup.Orientation = \_

wdOrientLandscape

With ActiveDocument.PageSetup

a=InputBox("Введите размер левого поля")

b=InputBox("Введите размер правого поля")

c=InputBox("Введите размер верхнего поля")

d=InputBox("Введите размер нижнего поля")

.LeftMargin = CentimetersToPoints(a)

.RightMargin = CentimetersToPoints(b)

.TopMargin = CentimetersToPoints(c)

.BottomMargin = CentimetersToPoints(d)

End With

If ActiveWindow.View.SplitSpecial <> wdPaneNone Then

ActiveWindow.Panes(2).Close

End If

If ActiveWindow.ActivePane.View.Type <> wdPrintView Then

ActiveWindow.ActivePane.View.Type = wdPrintView

End If

With Selection.PageSetup.TextColumns

.SetCount NumColumns:=2

.EvenlySpaced = True

.LineBetween = False

.Width = CentimetersToPoints(11.13)

End With

End Sub

**Вывод**: Научился программировать в Word на VBA, писать макросы

**Контрольные вопросы**

**1**. Макрос (макрокоманда) – последовательность команд, имеющая имя. С помощью макросов можно автоматизировать действия, часто выполняемые пользователем. Для написания макросов используется язык Visual Basic for Application (VBA), встроенный в пакет Microsoft Office.

**2**. Проект – это основная программа, которая объединяет и инициализирует все подключаемые модули, библиотеки и т.п. Обычно содержит описания объектов и интерфейса, вызовы процедур из подключенных модулей. В состав проекта входят следующие компоненты:

1) модули – содержат процедуры макросов ;

2) модули класса – содержат описания пользовательских классов;

3) формы – содержат описания форм и процедур обработки событий; позволяют организовать диалог с пользователем;

4) объекты Microsoft Word – делают доступными другие документы;

5) ссылки – подключение библиотек; позволяют сделать объект другого приложения доступным в программе.

Наиболее часто используемые компоненты – это модули и формы.

В VBA существует 2 типа проектов :

1) Normal – проект сохранен вместе с шаблоном Normal.dot; содержащиеся в нем макросы доступны для любого документа Word;

2) Project – хранит только макросы данного документа.

**3**. Язык Visual Basic построен на синтаксисе языка Basic и поддерживает все его конструкции.

Компилятор языка нечувствителен к регистру клавиатуры. Каждый оператор записывается с новой строки. При необходимости записать два оператора на одной строке их разделяют двоеточием. Строки комментариев обозначаются командой REM или одинарной кавычкой.

**4.** При написании программы рекомендуется создать макрос в автоматическом режиме (Сервис- Макрос- Начать запись). Затем просмотреть сгенерированный текст в редакторе Visual Basic для определения объектов, методов и свойств, которые следует использовать при написании программы. При необходимости дополнительную информацию можно получить из справки в разделе Справка по Visual Basic.