Кафедра «Автоматизированные системы управления»

**Курсовая работа**

**по дисциплине ВТиП**

**Разработка программного модуля**

**Введение**

Целью данной курсовой работы является разработка программного модуля, с помощью которого можно задать размерность квадратной матрицы, заполнить матрицу случайными целыми числами от 0 до 6 и вычислить:

- сумму элементов, находящихся под главной диагональю,

- сумму элементов, составляющих главную диагональ.

Для разработки программы использован табличный процессор Excel и язык программирования Visual Basic for Application.

В первом разделе, который называется «Постановка задачи», приведена математическая модель задачи, описаны входные и выходные данные, описана обработка ошибок.

Во втором разделе, «Проектирование программного модуля», изображена структурная диаграмма программного модуля, разработана схема программного модуля и пользовательский интерфейс.

В разделе «Реализация программного модуля» описан код программы, произведено описание используемых операторов и функций.

Тестирование программного модуля приведено в четвертом разделе.

Кроме того, дано заключение и приведён список использованных источников.

**1. Постановка задачи**

**1.1 Математическая модель задачи**

Определение квадратной матрицы: квадратной матрицей n-го порядка называется матрица, состоящая из m строк и m столбцов. Главной диагональю квадратной матрицы называется диагональ, составленная из элементов a11 a22… amm .

Способ нахождения суммы элементов квадратной матрицы, лежащих под главной диагональю:

Необходимый результат получается при суммировании элементов в следующем порядке: а21 + а31 + а32 +…+ аi1 + ai2 + ai3 + ai(j-1) – суммируются элементы, начиная со второй строки, и, при увеличении строки на 1, число столбцов, содержащих нужные элементы, также увеличивается на 1. В кратком виде этот цикл выглядит так: i=f…m, где начальное f=2; j=1…(m-b), где начальное b=m-1. При каждой последующей итерации значение f увеличивается на 1, а значение b на 1 уменьшается.

Сумму элементов диагонали матрицы можно получить при суммировании элементов, лежащих на пересечении одинаковых строки и столбца, т.е. если i=j.

**1.2 Входные данные**

В данном курсовом проекте к входным данным относится размерность квадратной матрицы (значение m).

Требования к входным данным:

- значение m должно вводиться в числовой форме;

- значение m должно быть целым, положительным числом больше нуля.

**1.3 Выходные данные**

К выходным данным относятся:

- вывод заполненной матрицы на форму;

- вывод заполненной матрицы на лист Excel;

- сумма элементов, находящихся под главной диагональю;

- сумма элементов, составляющих главную диагональ.

Требования к выходным данным:

- выходные данные выводятся в числовом виде.

**1.4 Обработка ошибок**

При выполнении программного модуля произведена обработка следующих ошибок:

- в поле ввода размерности матрицы вводится нечисловое значение. В этом случае появляется сообщение об ошибке, которое имеет вид, представленный на рисунке 1:

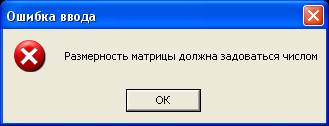


Рисунок 1 - Сообщение об ошибке, в случае нечислового ввода размерности матрицы

- в поле ввода вводится дробное числовое значение. В этом случае появляется сообщение об ошибке, которое имеет вид, представленный на рисунке 2:

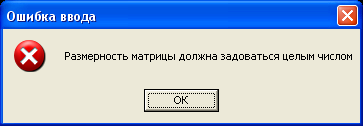


Рисунок 2 - Сообщение об ошибке, в случае ввода дробного значения размерности матрицы

- в поле ввода вводится отрицательное число, либо число равное нулю. В этом случае появляется сообщение об ошибке, которое имеет вид, представленный на рисунке 3:

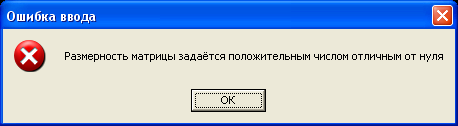


Рисунок 3 - Сообщение об ошибке, в случае ввода отрицательного, либо нулевого значения размерности матрицы

После уведомления пользователя об ошибке поле ввода очищается, и курсор возвращается в это поле.

**2. Проектирование программного модуля**

**2.1 Структурная диаграмма программного модуля**

Программа имеет структуру, показанную на рисунке 4:

UserForm1

UserForm\_Initialize()

CommandButton1\_

Click( )

CommandButton2\_

Click( )

CommandButton3\_

Click( )

CommandButton4\_

Click( )

CommandButton5\_

Click( )

Option Button1\_ Click()

Option Button2\_ Click()

Рисунок 4 - Структурная диаграмма программного модуля

В данной диаграмме UserForm\_Initialize() – процедура инициализации пользовательской формы. CommandButton1\_Click() – процедура, срабатывающая при нажатии кнопки «Заполнить матрицу», которая производит проверку правильности ввода размерности матрицы, заполнение матрицы и вывод её на пользовательскую форму. CommandButton2\_Сlick() – процедура, срабатывающая при нажатии кнопки «Очистить», которая очищает форму. CommandButton3\_Сlick() – процедура, срабатывающая при нажатии кнопки «Выйти», позволяющая пользователю завершить работу с программой. CommandButton4\_Click() – процедура, срабатывающая при нажатии кнопки «О программе», которая даёт краткие сведения о программе и её разработчике. CommandButton5\_Сlick() – процедура, срабатывающая при нажатии кнопки «Работать с Excel», позволяющая пользователю переключиться на работу с листом Excel. OptionButton1\_Click() – процедура-переключатель, при выборе которого вычисляется сумма элементов, находящихся под главной диагональю. OptionButton2\_Click() – процедура-переключатель, при выборе которого вычисляется сумма элементов, составляющих главную диагональ.

**2.2 Разработка схемы программного модуля и ее описание**

Блок-схема процедуры заполнения квадратной матрицы представлена на рисунке 5:

НАЧАЛО

m = TextBox1.Text

IsNumeric(m) = False

m <= 0

m <> Int(m)

ReDim a(1 To m, 1 To m)

i=1 To m

j=1 To m

a(i, j) = Int((7 \* Rnd) + 0)

A

B

Да

Да

Да

Нет

Нет

Нет

к странице 9

к странице 9

1

2

3

4

5

6

7

8

Вывод a(i,j) на ListBox

КОНЕЦ

B

На страницу 8

A

на страницу 8

9

Рисунок 5 – Схема программного модуля (Заполнение матрицы)

Описание блок-схемы:

1 – ввод размера массива;

2 – проверка того, что введённый размер массива является числом;

3 – проверка того, что введённый размер массива является положительным числом, отличным от нуля;

4 – проверка того, что введённый размер массива является целым числом;

5 – задание динамического массива;

6 – цикл, который пробегает значения строк от 1 до заданного размера массива, с шагом равным по умолчанию 1;

7 – цикл, который пробегает значения столбцов от 1 до заданного размера массива, с шагом равным по умолчанию 1;

8 – тело цикла, которое заполняет массив случайными числами от 0 до 6;

9 – вывод результата в поле, предназначенное для вывода квадратной матрицы.

Блок-схема процедуры для первого переключателя представлена на рисунке 6:

НАЧАЛО

summa1 = 0; f = 2

b = m - 1

i = f To m

j = 1 To m - b

summa1 = summa1 + a(i, j)

f = f + 1;

b = b - 1

Вывод результата

КОНЕЦ

1

2

3

4

5

6

Рисунок 6 – Схема программного модуля (первый переключатель)

Описание блок-схемы:

1 – задание начальных параметров для вычисления суммы элементов матрицы;

2 – цикл, который пробегает значения строк от f до заданного размера массива, с шагом равным по умолчанию 1;

3 – цикл, который пробегает значения столбцов от 1 до m-b, с шагом равным по умолчанию 1;

4 – вычисление суммы элементов, лежащих под главной диагональю;

5 – увеличение параметра f на 1 и уменьшение параметра b на 1, после окончания цикла для столбцов (блок 3), и переход к циклу для строк (блок 2);

6 – вывод результата в поле вывода суммы элементов.

Блок-схема процедуры для второго переключателя представлена на рисунке 7:

НАЧАЛО

summa2 = 0

i = 1 To m

j = 1 To m

summa2 = summa2 + a(i, j)

Вывод результата

КОНЕЦ

i=j

Да

Нет

1

2

3

4

5

6

Рисунок 7 – Схема программного модуля (второй переключатель)

Описание блок-схемы:

1 – задание начальных параметров для вычисления суммы элементов матрицы;

2 – цикл, который пробегает значения строк от 1 до заданного размера массива, с шагом равным по умолчанию 1;

3 – цикл, который пробегает значения столбцов от 1 до заданного размера массива, с шагом равным по умолчанию 1;

4 – проверка условия i = j. В случае выполнения данного условия происходит переход к блоку 5, в противном случае – к блоку 3;

5 – вычисление суммы элементов, составляющих главную диагональю;

6 – вывод результата в поле вывода суммы элементов.

Блок-схема процедуры для работы с листом Excel представлена на рисунке 8:

НАЧАЛО

m = TextBox1.Text

IsNumeric(m) = False

m <= 0

m <> Int(m)

ReDim a(1 To m, 1 To m)

i=1 To m

j=1 To m

a(i, j) = Int((7 \* Rnd) + 0)

A

B

Да

Да

Да

Нет

Нет

Нет

к странице 13

Cells(i + 5, j) = a(i, j)

к странице 13

1

2

3

4

5

6

7

8

9

B

На страницу12

A

на страницу12

summa1 = 0; f = 2

b = m - 1

i = f To m

j = 1 To m - b

summa1 = summa1 + a(i, j)

f = f + 1;

b = b - 1

Вывод результата на лист Excel

summa2 = 0

i = 1 To m

j = 1 To m

summa2 = summa2 + a(i, j)

i=j

Да

Нет

D

к странице 14

С

к странице 14

10

11

12

13

14

15

16

17

18

19

20

21

Вывод результата на лист Excel

КОНЕЦ

D

на страницу13

C

на страницу 13

Рисунок 8 – Схема программного модуля (Работа с листом Excel)

Описание блок-схемы:

1 – ввод размера массива;

2 – проверка того, что введённый размер массива является числом;

3 – проверка того, что введённый размер массива является положительным числом, отличным от нуля;

4 – проверка того, что введённый размер массива является целым числом;

5 – задание динамического массива;

6 – цикл, который пробегает значения строк от 1 до заданного размера массива, с шагом равным по умолчанию 1;

7 – цикл, который пробегает значения столбцов от 1 до заданного размера массива, с шагом равным по умолчанию 1;

8 – тело цикла, которое заполняет массив случайными числами от 0 до 6;

9 – вывод массива на лист Excel;

10 – задание начальных параметров для вычисления суммы элементов матрицы, расположенных под главной диагональю;

11 – цикл, который пробегает значения строк от f до заданного размера массива, с шагом равным по умолчанию 1;

12 – цикл, который пробегает значения столбцов от 1 до m-b, с шагом равным по умолчанию 1;

13 – вычисление суммы элементов, лежащих под главной диагональю;

14 – увеличение параметра f на 1 и уменьшение параметра b на 1, после окончания цикла для столбцов (блок 12), и переход к циклу для строк (блок 11);

15 – вывод суммы элементов, лежащих под главной диагональю на лист Excel;

16 – задание начальных параметров для вычисления суммы элементов матрицы, составляющих главную диагональ;

17 – цикл, который пробегает значения строк от 1 до заданного размера массива, с шагом равным по умолчанию 1;

18 – цикл, который пробегает значения столбцов от 1 до заданного размера массива, с шагом равным по умолчанию 1;

19 – проверка условия i = j. В случае выполнения данного условия происходит переход к блоку 5, в противном случае – к блоку 3;

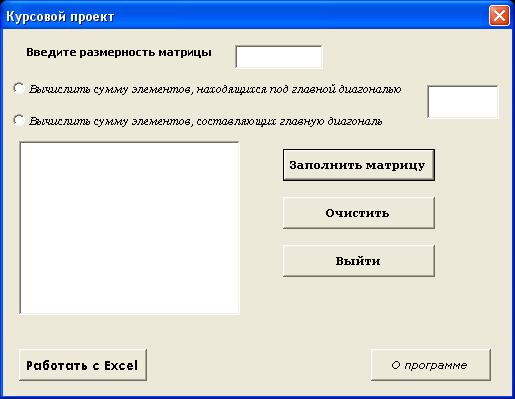
20 – вычисление суммы элементов, составляющих главную диагональю;

21 – вывод суммы элементов, составляющих главную диагональ на лист Excel.

2.3 Разработка пользовательского интерфейса.

Пользовательский интерфейс (ПИ) программы - это совокупность элементов, позволяющих пользователю программы управлять ее работой и получать требуемые результаты, т.е. это диалог между компьютером и пользователем.

Интерфейс для программного модуля, разработанного в данном курсовом проекте представлен на рисунке 9:



2

1

5

6

7

10

3

4

8

9

Рисунок 9 - Вид пользовательской формы: 1 – поле для ввода размерности квадратной матрицы; 2 – поле для вывода суммы элементов матрицы, в зависимости от выбранного переключателя; 3 – переключатель, при выборе которого вычисляется сумма элементов матрицы, находящихся под главной диагональю; 4 – переключатель, при выборе которого вычисляется сумма элементов матрицы, составляющих главную диагональ; 5 – кнопка, при нажатии которой происходит заполнение матрицы; 6 – кнопка, которая осуществляет очистку всех полей формы; 7 – кнопка, осуществляющая выход из программы; 8 – кнопка, при нажатии которой появляется краткая информация о программе; 9 – кнопка, позволяющая пользователю переключиться на работу с листом Excel; 10 - поле для вывода заполненной матрицы

**3. Реализация программного модуля**

**3.1 Код программы**

Dim summa1 As Double

Dim summa2 As Double

Dim a() As Double

Dim m As Variant

‘задаём начальные параметры при инициализации формы:

Private Sub UserForm\_Initialize()

Application.Visible = False

UserForm1.Caption = "Курсовой проект"

CommandButton1.Default = True

TextBox1.SetFocus

End Sub

‘процедура заполнения матрицы:

Private Sub CommandButton1\_Click()

m = TextBox1.Text

If IsNumeric(TextBox1.Text) = False Then

MsgBox "Размерность матрицы должна задаваться числом", 16, "Ошибка ввода"

TextBox1.Text = ""

TextBox1.SetFocus

Exit Sub

End If

If m <= 0 Then

MsgBox "Размерность матрицы задаётся положительным числом отличным от нуля ", 16,

"Ошибка ввода"

TextBox1.Text = ""

TextBox1.SetFocus

Exit Sub

End If

m = CDbl(m)

If m <> Int(m) Then

MsgBox " Размерность матрицы должна задаваться целым числом ", 16, " Ошибка ввода "

TextBox1.Text = ""

TextBox1.SetFocus

Exit Sub

End If

ReDim a(1 To m, 1 To m)

For i = 1 To m

For j = 1 To m

a(i, j) = Int((7 \* Rnd) + 0)

Next j

Next i

With ListBox1

ColumnCount = m

List = a

End With

End Sub

'процедура очистки пользовательской формы:

Private Sub CommandButton2\_Click()

OptionButton1.Value = False

OptionButton2.Value = False

TextBox1.Text = ""

TextBox2.Text = ""

ListBox1.Clear

TextBox1.SetFocus

End Sub

'процедура выхода из программы:

Private Sub CommandButton3\_Click()

UserForm1.Hide

Application.Quit

End Sub

' вызов краткой информации о программе:

Private Sub CommandButton4\_Click()

MsgBox "Программа для заполнения случайными числами" & Chr(13) & \_

"от 0 до 6 квадратной матрицы, размерностью" & Chr(13) & \_

"задаваемой пользователем, и вычисления суммы" & Chr(13) & \_

"элементов матрицы, в зависимости от выбрано-" & Chr(13) & \_

"го переключателя." & Chr(13) & \_

" Разработчик: Логунов А.П..", 48, "О программе"

End Sub

'процедура вычисления суммы элементов, расположенных под главной диагональю:

Private Sub OptionButton1\_Click()

summa1 = 0

f = 2

b = m - 1

For i = f To m

For j = 1 To m - b

summa1 = summa1 + a(i, j)

Next j

f = f + 1

b = b - 1

Next i

TextBox2.Text = summa1

End Sub

'процедура вычисления суммы элементов, составляющих главную диагональ:

Private Sub OptionButton2\_Click()

summa2 = 0

For i = 1 To m

For j = 1 To m

If i = j Then

summa2 = summa2 + a(i, j)

End If

Next j

Next i

TextBox2.Text = summa2

End Sub

'процедура для работы с Excel:

Private Sub CommandButton5\_Click()

Application.Visible = True

Cells.Select

Selection.ClearContents

Range("A1").Select

UserForm1.Hide

m = InputBox("Задайте размерность матрицы", "Окно ввода")

If IsNumeric(m) = False Then

MsgBox ""Размерность матрицы должна задаваться числом", 16, "Ошибка ввода"

Exit Sub

End If

If m <= 0 Then

MsgBox "Размерность матрицы задаётся положительным числом отличным от нуля ", 16,

"Ошибка ввода"

Exit Sub

End If

m = CDbl(m)

If m <> Int(m) Then

MsgBox " Размерность матрицы должна задаваться целым числом ", 16, " Ошибка ввода "

Exit Sub

End If

Cells(5, 1) = "Матрица размерностью n=" & m & ":"

ReDim a(1 To m, 1 To m)

For i = 1 To m

For j = 1 To m

a(i, j) = Int((7 \* Rnd) + 0)

Cells(i + 5, j) = a(i, j)

Next j

Next i

summa1 = 0

f = 2

b = m - 1

For i = f To m

For j = 1 To m - b

summa1 = summa1 + a(i, j)

Next j

f = f + 1

b = b - 1

Next i

Cells(2, 1) = "Сумма элементов под главной диагональю =" & summa1

summa2 = 0

For i = 1 To m

For j = 1 To m

If i = j Then

summa2 = summa2 + a(i, j)

End If

Next j

Next i

Cells(3, 1) = " Сумма элементов составляющих главную диагональ =" & summa2

Select Case MsgBox("Вернуться к UserForm?", vbYesNo, "Окно возврата")

Case vbYes

Application.Visible = False

TextBox1.SetFocus

UserForm1.Show

Case vbNo

End Select

End Sub

**3.2 Описание использованных операторов и функций**

Dim Имя\_переменной As Тип\_переменной – синтаксис описания типа переменной;

Private – указывает, что процедура Sub доступна для всех других процедур только того модуля, в котором она описана;

If Условие Then [Инструкция] [Else Инструкции\_else] – оператор условного перехода. Если условие принимает значение True, то выполняется инструкция Then, если False, то выполняется инструкция\_else. Ветвь Else является необязательной;

IsNumeric(переменная) – функция, проверяющая является ли переменная числовым значением;

MsgBox (сообщение, [кнопка], [заголовок])- выводит на экран диалоговое окно, содержащее сообщение;

CDbl() – функция преобразования считываемых данных в числовой формат типа Double, т.к. числа, вводимые в текстовую область формы, воспринимаются как текст, а не как число.

Int() – функция, которая возвращает целые числовые значения;

ReDim <имя массива>(<задаётся размерность массива>) – функция задания динамического массива;

For Счетчик = Начало To Конец [Step Шаг]

[Инструкции]

Next Счетчик – повторяет выполнение группы инструкций, пока Счетчик изменяется от начального значения до конечного с указанным шагом. Если шаг не указан, то он полагается равным 1;

Rnd – функция, которая служит для генерации случайных чисел;

With Объект

[инструкции]

End With – позволяет выполнить последовательность инструкций над Объектом не повторяя его имени;

ColumCount – устанавливает число столбцов в списке;

Clear – очистка;

SetFocus – возвращает курсор в указанное поле;

InputBox () – выводит на экран диалоговое окно, содержащее сообщение и поле ввода, устанавливает режим ожидания ввода текста пользователем или нажатия кнопки, а затем возвращает значение типа String, содержащее текст, введенный в поле;

Select Case Выражение

Case список выражений

[инструкции]

Case Else

[инструкции Else]

End Select – выполняет одну из нескольких групп инструкций в зависимости от некоторого выражения.

**4. Тестирование программного модуля**

Ниже приведён пример работы программного модуля. Для этого ввели в поле ввода размерность матрицы равную 5. При нажатии кнопки «Заполнить матрицу» в окне вывода появилось квадратная матрица пятого порядка, заполненная случайными числами от 0 до 6, что представлено на рисунке 10:

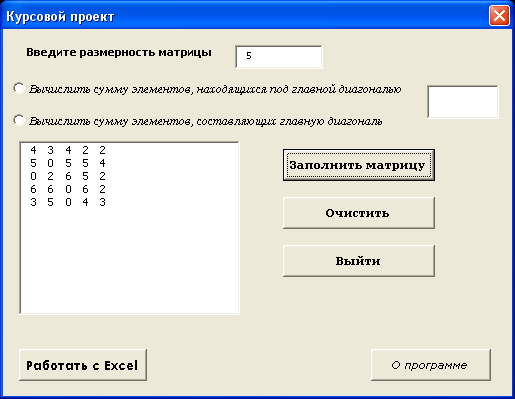


Рисунок 10 - Вид пользовательской формы с заполненной матрицей

При выборе первого переключателя определяется сумма элементов, находящихся под главной диагональю матрицы. При выборе второго переключателя – сумма элементов матрицы, составляющих главную диагональ.

При нажатии кнопки «о программе» появляется сообщение с краткой информацией о программе.

При нажатии кнопки «Работать с Excel» появляется диалоговое окно, в котором задаётся размерность матрицы.

При нажатии кнопки «Оk» на листе Excel появляется результат, который представлен на рисунке 11:

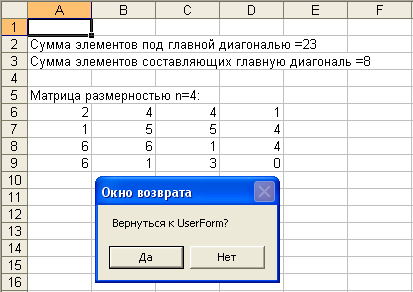


Рисунок 11 – Лист Excel с результатом

Диалоговое окно «окно возврата» позволяет пользователю вернуться к работе с пользовательской формой.

Как видно из приведенного примера программа является полностью работоспособной.

**Заключение**

В данном курсовом проекте с помощью языка программирования Visual Basic for Application был разработан программный модуль, который позволяет: задавать квадратную матрицу и программно заполнять её случайными числами от 0 до 6; вычислять сумму элементов находящихся под главной диагональю; вычислять сумму элементов составляющих главную диагональ. Был предусмотрен режим работы с листом Excel.

В программе произведена обработка возможных ошибок, что облегчает пользователю работу с программой.

В ходе тестирования программа показала себя полностью работоспособной. Следовательно, поставленную задачу можно считать выполненной.

**Список использованных источников**

1. Гарнаев А.Ю. Самоучитель VBA. – СПб.: БХВ – Санкт-Петербург, 2000. – 512 с.

2. Гарнаев А.Ю. Самоучитель VBA. – СПб, БХВ – Санкт-Петербург, 2002.

3. MS OfficeXP: Разработка приложений / Под редакцией Ф.А. Новикова. СПб.: БХВ – Санкт-Петербург, 2003.