ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени федеральное государственное

бюджетное образовательное учреждение высшего образованияa

Московский Технический Университет Связи и Информатики

(МТУСИ)



Кафедра информатики

Дисциплина Информатика

Лабораторная работа № 05-01

«Модульная структура приложений и стандартные модули в Visual Basic»

Вариант № 7

Выполнил: Калининский Д.С.

Студент 1-ого курса ОТФ 2

Группы БИН1703

Преподаватель: Кравченко О.М.

Москва 2018

Содержание

[**1 Задание** 3](#_Toc511995185)

[**2 Форма проекта** 5](#_Toc511995186)

[**3 Схемы алгоритмов процедур и программы** 6](#_Toc511995187)

[**4 Таблица свойств объектов** 10](#_Toc511995188)

[**5 Текст программы** 12](#_Toc511995189)

[**6 Результат тестирования программы** 16](#_Toc511995190)

# **1 Задание**

**1)** Изучите основные понятия и определения модульной структуры приложений VB, использующих несколько форм, а также способы работы со стандартными модулями **(Тема 1).**

**2)** Изучите общее задание в п. 1.5-2 и выберите вариант задания в соответствии с указанием преподавателя, а также по указанию преподавателя выберите варианты реализации проекта: со стандартным модулем и двумя формами и вводом с клавиатуры; со стандартным модулем и двумя формами и вводом из файла.

**3)** Проведите формализацию задачи, для чего:

• определите перечень исходных данных и предусмотрите для них проверку правильности ввода;

• представьте смысловые данные, которые будут использоваться в ходе решения задачи в виде переменных и массивов, присвоив им соответствующие имена;

• определите размерности и типы используемых данных;

• представьте выходные данные в виде массивов;

• приведите геометрическую иллюстрацию решения задачи;

• при необходимости приведите расчетные формулы, которые будут использоваться для преобразования исходных данных в результаты.

**4)** Разработайте интерфейс пользователя, то есть:

• разработайте форму, предназначенную для ввода исходных данных, как с клавиатуры, так и из текстового файла, и для отображения входных данных на форме;

• разработайте выходную форму, предназначенную для вывода результатов решения задачи, как на форму, так и для записи в файл.

**5)** Составьте схемы алгоритмов решения функциональных.

**6)** Создайте программный код проекта, реализующий поставленную задачу.

**7)** Подготовьте тестовые наборы исходных данных, соответствующие смысловой направленности решаемой задачи как для ввода с клавиатуры, так и из текстового файла. Формат текстового файла должен быть задан преподавателем (см. примеры выполнения задания п. 1.5.4), а сам текстовый файл с расширением \*.txt должен быть создан студентом средствами MS Windows, например, в Блокноте.

**8)** Выполните проект и получите решение.

**9)** Обоснуйте правильность полученных результатов.

Таблица 1 – Индивидуальный вариант задания

|  |  |
| --- | --- |
| Вариант | Задание |
| 7) | Ремонтное предприятие выполняет 6 видов работ. Для каждого вида известно название и стоимость. К предприятию обратились n заказчиков с предложениями выполнить определенное количество работ каждого вида. Составить алгоритм и программу, которые для каждого заказчика определяют сумму договора с ремонтным предприятием, а также организуют формирование списка заказчиков в порядке убывания сумм договоров. |

# **2 Формы проекта**

На рисунке 1 и 2 изображена формы проекта **Проект 5.1.**

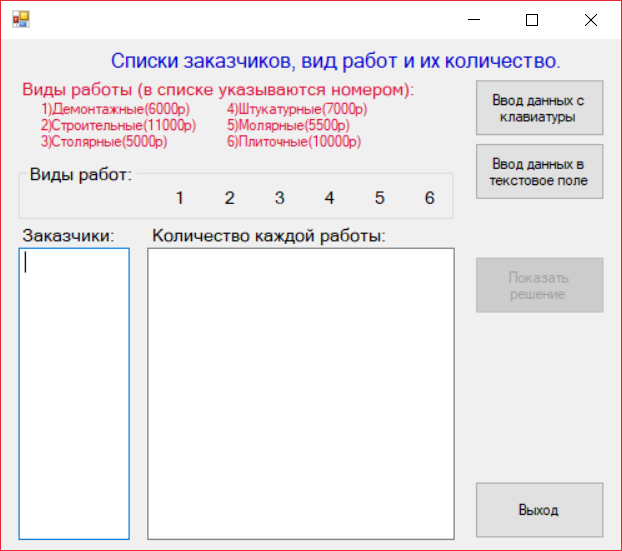


Рисунок 1 – Первая форма проекта

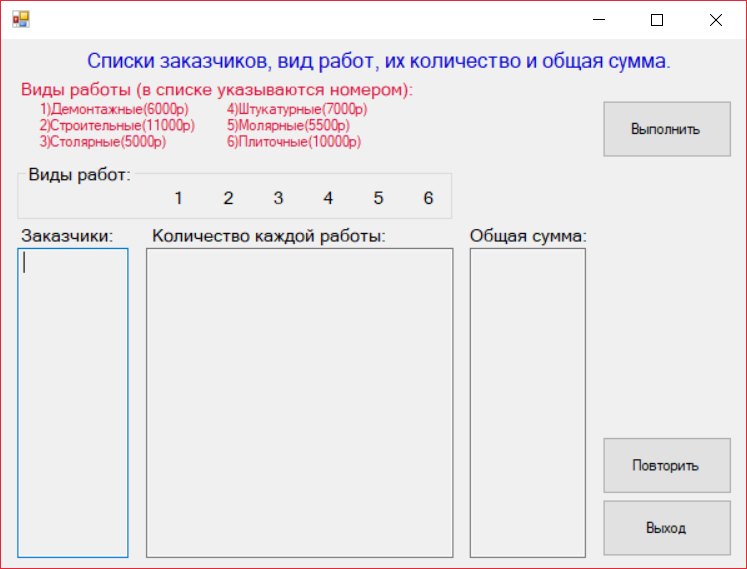


Рисунок 2 – Вторая форма проекта

# **3 Схемы алгоритмов процедур и программы**

Схема алгоритма процедуры unic() представлена на рисунке 3, схема алгоритма процедуры sort() представлена на рисунке 4, схема алгоритма процедуры izm() представлена на рисунке 5, схема алгоритма событийной процедуры Form1.Btn1\_Click() на рисунке 6, схема алгоритма событийной процедуры Form1.Btn2\_Click() на рисунке 7 и схема алгоритма событийной процедуры Form2.Btn1\_Click() на рисунке 8.

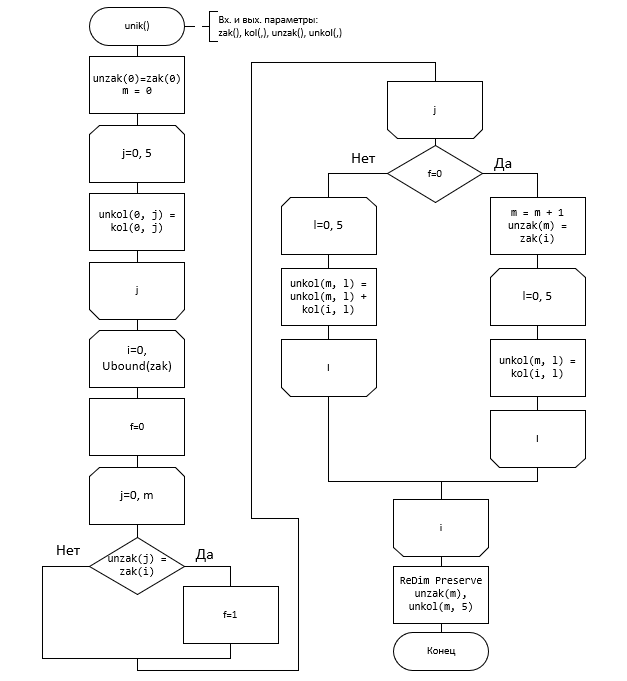


Рисунок 3 – Схема алгоритма процедуры unic()

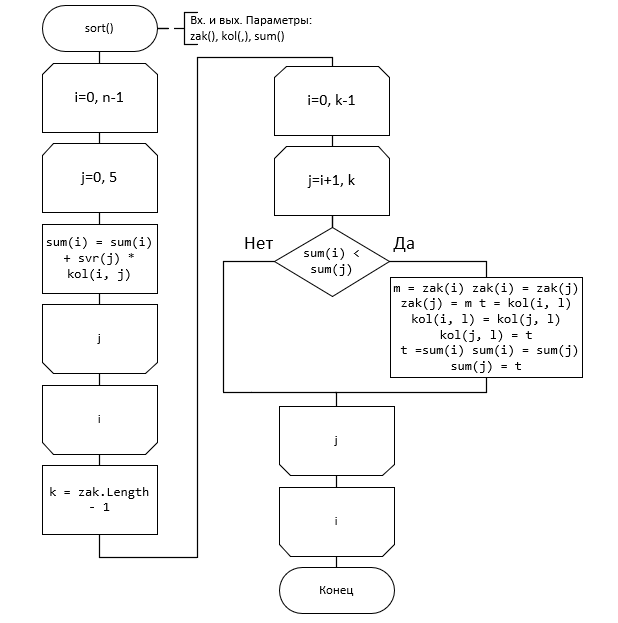


Рисунок 4 - Схема алгоритма процедуры sort()

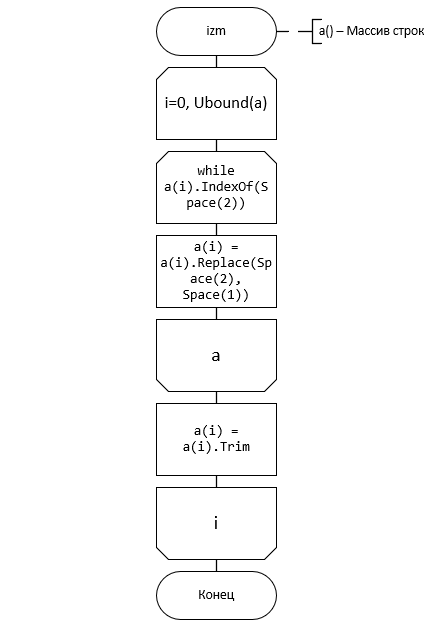


Рисунок 5 - Схема алгоритма процедуры izm()

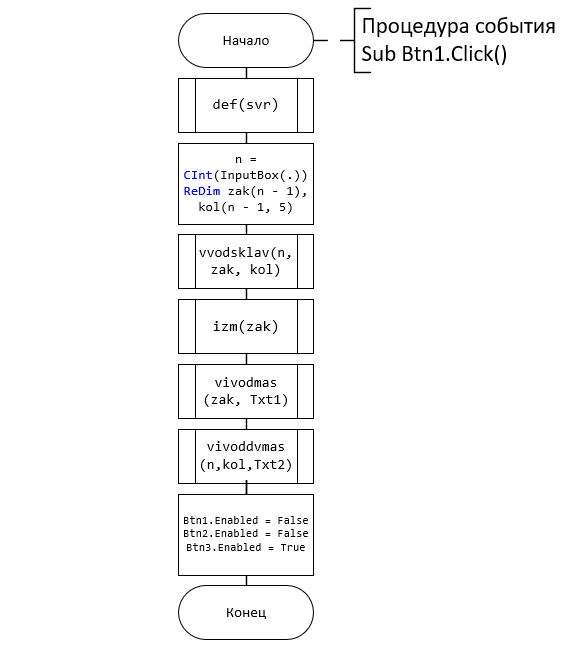


Рисунок 6 - Схема алгоритма процедуры события Form1 Sub Btn1\_Click()

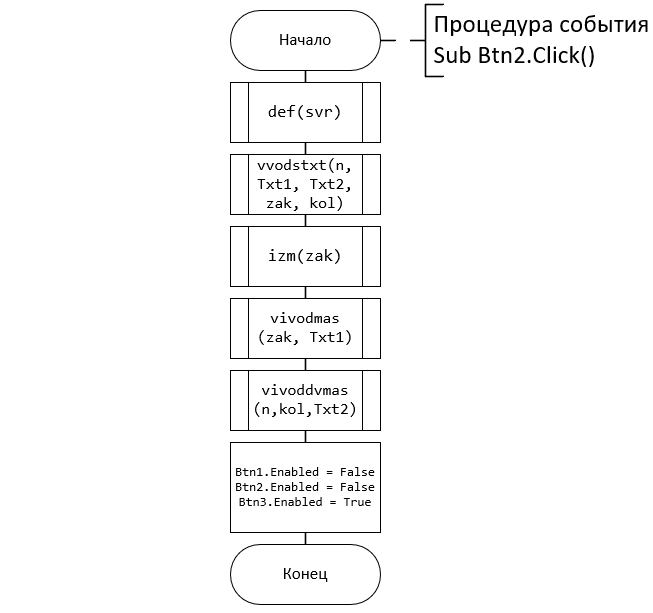


Рисунок 7 - Схема алгоритма процедуры события Form1 Sub Btn2\_Click()

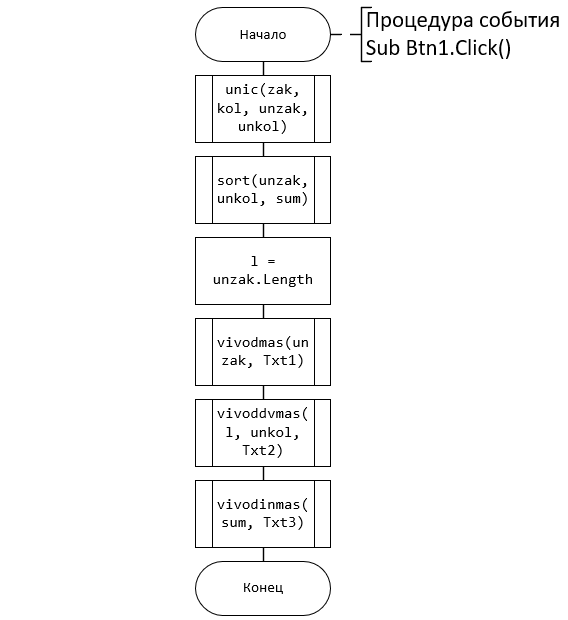


Рисунок 8 - Схема алгоритма процедуры события Form2 Sub Btn1\_Click()

# **4 Таблица свойств объектов**

В таблице 2 и 3 можно узнать свойства всех объектов программы.

Таблица 2 – Свойства объектов

|  |  |  |
| --- | --- | --- |
| **Form1** | | |
| **Имя объекта** | **Свойства** | **Значение свойств** |
| **lbl1** | ForeColor | MediumBlue |
| Font | Microsoft Sans Serif; 12pt |
| Text | Списки заказчиков, вид работ и их количество. |
| **lbl2** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | Задание: |
| **lbl3** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | 1)Демонтажные(6000р) 2)Строительные(11000р) 3)Столярные(5000р) |
| **lbl4** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | 4)Штукатурные(7000р) 5)Молярные(5500р) 6)Плиточные(10000р) |
| **lbl5** | Font | Microsoft Sans Serif; 10,2pt |
| Text | 1 2 3 4 5 6 |
| **lbl6** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Заказчики: |
| **lbl7** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Количество каждой работы: |
| **GroupBox1** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Виды работ: |
| **txt1** | Name | txt1 |
| MultiLines | True |
| **txt2** | Name | txt2 |
| Multilines | True |
| **Btn1** | Text | Ввод данных с клавиатуры |
| **Btn2** | Text | Ввод данных в текстовое поле |
| **Btn3** | Text | Показать решение |
| **Btn4** | Text | Выход |

Таблица 3 – Свойства объектов

|  |  |  |
| --- | --- | --- |
| **Form2** | | |
| **Имя объекта** | **Свойства** | **Значение свойств** |
| **lbl1** | ForeColor | MediumBlue |
| Font | Microsoft Sans Serif; 12pt |
| Text | Списки заказчиков, вид работ, их количество и общая сумма. |
| **lbl2** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | Задание: |
| **lbl3** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | 1)Демонтажные(6000р) 2)Строительные(11000р) 3)Столярные(5000р) |
| **lbl4** | ForeColor | Crimson |
| Font | Microsoft Sans Serif; 10,2pt |
| Text | 4)Штукатурные(7000р) 5)Молярные(5500р) 6)Плиточные(10000р) |
| **lbl5** | Font | Microsoft Sans Serif; 10,2pt |
| Text | 1 2 3 4 5 6 |
| **lbl6** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Заказчики: |
| **lbl7** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Количество каждой работы: |
| **lbl8** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Общая сумма: |
| **GroupBox1** | Font | Microsoft Sans Serif; 10,2pt |
| Text | Виды работ: |
| **txt1** | Name | txt1 |
| MultiLines | True |
| **txt2** | Name | txt2 |
| Multilines | True |
| **txt3** | Name | txt3 |
| Multilines | True |
| **Btn1** | Text | Выполнить |
| **Btn2** | Text | Повторить |
| **Btn3** | Text | Выход |

# **5 Текст программы**

Module Module1

Public svr(5) As Integer ' Сумма вида работы

Public n As Integer ' Количество заказчиков

Public zak() As String ' Заказчик

Public kol(,) As Integer ' Количество работ ведённое

Public sum() As Integer ' Сумма

Public unzak() As String

Public unkol(,) As Integer

Public Sub def(ByRef svr() As Integer)

svr(0) = 6000 : svr(1) = 11000 : svr(2) = 5000

svr(3) = 7000 : svr(4) = 5500 : svr(5) = 10000

End Sub

Public Sub unic(ByRef zak() As String, ByRef kol(,) As Integer,

ByRef unzak() As String, ByRef unkol(,) As Integer)

Dim i, j, l, f, m As Integer

Dim proverka As String = ""

unzak(0) = zak(0) : m = 0

For j = 0 To 5

unkol(0, j) = kol(0, j)

Next j

For i = 0 To UBound(zak)

f = 0

For j = 0 To m

If unzak(j) = zak(i) Then f = 1

Next j

If f = 0 Then

m = m + 1

unzak(m) = zak(i)

For l = 0 To 5

unkol(m, l) = kol(i, l)

Next l

Else

For l = 0 To 5

unkol(m, l) = unkol(m, l) + kol(i, l)

Next l

End If

Next i

ReDim Preserve unzak(m), unkol(m, 5)

End Sub

Public Sub sort(ByRef zak() As String, ByRef kol(,) As Integer,

ByRef sum() As Integer)

Dim i, j, k, t, l As Integer, m As String

For i = 0 To n - 1

For j = 0 To 5

sum(i) = sum(i) + svr(j) \* kol(i, j)

Next j

Next i

k = zak.Length - 1

For i = 0 To k - 1

For j = i + 1 To k

If sum(i) < sum(j) Then

m = zak(i) : zak(i) = zak(j) : zak(j) = m

t = kol(i, l) : kol(i, l) = kol(j, l) : kol(j, l) = t

t = sum(i) : sum(i) = sum(j) : sum(j) = t

End If

Next j

Next i

End Sub

Public Sub vvodsklav(ByVal n As Integer, ByRef zak() As String,

kol(,) As Integer)

Dim i As Integer

For i = 0 To n - 1

zak(i) = InputBox("Заказчик")

kol(i, 0) = CInt(InputBox("Количество демонтажных работ"))

kol(i, 1) = CInt(InputBox("Количество строительных работ"))

kol(i, 2) = CInt(InputBox("Количество столярных работ"))

kol(i, 3) = CInt(InputBox("Количество штукатурных работ"))

kol(i, 4) = CInt(InputBox("Количество молярных работ"))

kol(i, 5) = CInt(InputBox("Количество плиточных работ"))

Next i

End Sub

Public Sub vvodstxt(ByRef n As Integer, ByVal T1 As TextBox,

ByVal T2 As TextBox, ByRef zak() As String,

ByRef kol(,) As Integer)

Dim i, j As Integer, s, f(), m() As String

n = T1.Lines.Length

zak = T1.Text.Split(vbNewLine)

s = T2.Text : f = s.Split(vbNewLine)

For i = 0 To n - 1

Do While f(i).IndexOf(Space(2)) >= 0

f(i) = f(i).Replace(Space(2), Space(1))

Loop

f(i) = f(i).Trim

m = f(i).Split(Space(1))

For j = 0 To 5

kol(i, j) = m(j)

Next j

Next i

End Sub

Public Sub izm(ByRef a() As String)

Dim i As Integer

For i = 0 To UBound(a)

Do While a(i).IndexOf(Space(2))

a(i) = a(i).Replace(Space(2), Space(1))

Loop

a(i) = a(i).Trim

Next i

End Sub

Public Sub vivoddvmas(ByVal n As Integer, ByVal a(,) As Integer, ByRef TB As TextBox)

Dim i, j As Integer

TB.Text = ""

For i = 0 To n - 1

TB.Text = TB.Text + Space(2)

For j = 0 To 5

If a(i, j) Mod 10 = 0 Then

TB.Text = TB.Text + CStr(a(i, j)) + Space(5)

Else

TB.Text = TB.Text + CStr(a(i, j)) + Space(4)

End If

Next j

TB.Text = TB.Text + vbNewLine

Next i

End Sub

Public Sub vivodmas(ByVal a() As String, ByRef TB As TextBox)

Dim i As Integer

TB.Text = ""

For i = 0 To UBound(a)

TB.Text = TB.Text + a(i) + vbNewLine

Next

End Sub

Public Sub vivodinmas(ByVal a() As Integer, ByRef TB As TextBox)

Dim i As Integer

TB.Text = ""

For i = 0 To UBound(a)

TB.Text = TB.Text + a(i) + vbNewLine

Next

End Sub

End Module

Public Class Form1

Private Sub Btn1\_Click(sender As Object, e As EventArgs) Handles Btn1.Click

def(svr)

n = CInt(InputBox("Введите количество заказчиков"))

ReDim zak(n - 1), kol(n - 1, 5)

vvodsklav(n, zak, kol)

vivodmas(zak, Txt1)

vivoddvmas(n, kol, Txt2)

Btn1.Enabled = False

Btn2.Enabled = False

Btn3.Enabled = True

End Sub

Private Sub Btn2\_Click(sender As Object, e As EventArgs) Handles Btn2.Click

def(svr)

vvodstxt(n, Txt1, Txt2, zak, kol)

izm(zak)

vivodmas(zak, Txt1)

vivoddvmas(n, kol, Txt2)

Btn1.Enabled = False

Btn2.Enabled = False

Btn3.Enabled = True

End Sub

Private Sub Btn3\_Click(sender As Object, e As EventArgs) Handles Btn3.Click

Form2.Show()

Me.Hide()

End Sub

Private Sub Btn4\_Click(sender As Object, e As EventArgs) Handles Btn4.Click

End

End Sub

End Class

Public Class Form2

Private Sub Btn1\_Click(sender As Object, e As EventArgs) Handles Btn1.Click

Dim l As Integer

unic(zak, kol, unzak, unkol)

sort(unzak, unkol, sum)

l = unzak.Length

vivodmas(unzak, Txt1)

vivoddvmas(l, unkol, Txt2)

vivodinmas(sum, Txt3)

End Sub

Private Sub Btn2\_Click(sender As Object, e As EventArgs) Handles Btn2.Click

Form1.Txt1.Text = ""

Form1.Txt2.Text = ""

Form1.Btn1.Enabled = True

Form1.Btn2.Enabled = True

Form1.Btn3.Enabled = False

Form1.Show()

Me.Hide()

End Sub

Private Sub Btn3\_Click(sender As Object, e As EventArgs) Handles Btn3.Click

End

End Sub

End Class

# **6 Результат тестирования программы**

На рисунке 9 показан результат тестирования программы **Проект 5.1.**

Рисунок 9 – Результат программ