VISUAL BASIC/
C# PROGRAMMING
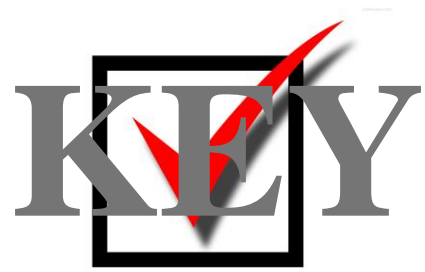## (330)

REGIONAL – 2015

Production Portion:

Program 1: File Statistics   _____ (500 points)

       TOTAL POINTS   _____ *(500 points)*

Judge/Graders: Please double check and verify all scores and answer keys!

General Instructions

XYZ, Inc. receives and processes submissions for composition contests and essay-based applications from across the nation. The first step in evaluating a submission is to verify that the submission meets the minimum requirements. For example, the maximum number of words allowed for a particular submission may be 500. This initial verification is currently a manual process and is proving to be overly time-consuming. XYZ, Inc. has contracted you to develop a program that will provide statistics about an ASCII text file that contains English composition to help streamline the verification process.

XYZ, Inc. has provided the following requirements:

1. The program must display the total number of ASCII characters in a file.
2. The program must display the total number of words contained in the file. A word is defined as any number of characters bounded by one or more spaces, the newline character, or the beginning or end of a string.
3. The program must display the total number of sentences contained in the file. A sentence is one or more words that end with a period, exclamation mark, or question mark.
4. The number of vowels contained in the file. A vowel can be any of the following: A, E, I, O, U, a, e, i, o, u.

You will have 90 minutes to complete your work.

Your name and/or school name should *not* appear on any work you submit for grading.

Copy your entire solution/project to the flash drive provided. You must submit your entire solution/project so that the graders may open your project to review the source code. You must ensure that the files required to run your program are present and will execute on the flash drive provided. Note that the flash drive letter may not be the same when the program is graded as it was when you created the program. It is recommended that you use relative paths rather than absolute paths to ensure that the program will run regardless of the flash drive letter. The graders will **not** compile or alter your source code to correct for this. Submissions that do not contain source code will **not be graded**.

Assumptions to make when taking this assessment:

- The input file(s) will contain only ASCII characters.
- The count of characters is any ASCII character, regardless if it is visible on the screen or in the file.
- A test input file will be available and will be named, "Student Data 330.txt".
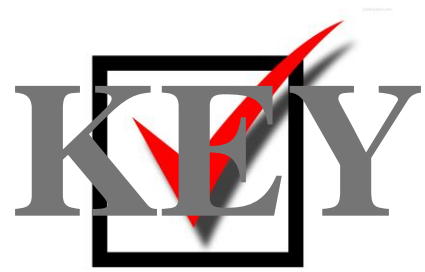
Technical Specifications:

1. **Solution and Project**
   a. Create a Visual Basic Windows Form Application named VB_330_ContestantNumber, where ContestantNumber is your BPA assigned contestant number. When naming your project, replace dashes (-) with the underscore (_). For example, if your BPA contestant number is 01-2345-6789, then your project name would be VB_330_01_2345_6789.
2. **User Interface**
   a. The user interface to be constructed is shown in Figure 1. Your application must be visually identical to the prototype shown in Figure 1.
   b. The Main Form
      i. The form is to be a Single Document Interface..
      ii. The name of the main form should be frmMain
      iii. The Form's caption must be set to "Submission Statistics: Contestant #ABC", where ABC is your BPA assigned contestant number (including dashes)
      iv. The StartPosition of the form must be set to CenterScreen
   c. User interface elements and default (design-time) property settings
      i. Exit Button
         1. The name of this button must be cmdExit
         2. The text property for this button must be set to "E&xit"
      ii. Clear Button
         1. The name of this button must be cmdClear
         2. The text property for this button must be set to "Clear"
         3. The Enabled property for this button must be set to False
      iii. Process Button
         1. The name of this button must be "cmdProcess"
         2. The text property for this button must be set to "Process'
         3. The Enabled property for this button must be set to False
      iv. Choose File Button
         1. The name of this button must be "cmdChooseFile"
         2. The text property for this button must be set to "Choose File"
      v. File Name Textbox
         1. The name of this textbox must be "txtFileName"
         2. The ReadOnly property for this textbox must be set to True
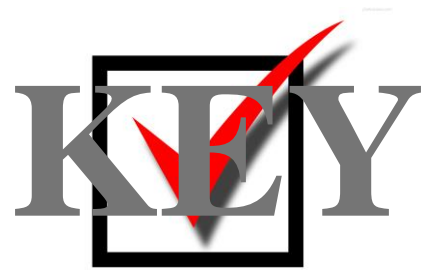
      vi.   Group Boxes
1. A group box called, "grpFileContents" must be created and must contain the textboxes and associated labels for: Characters, words, sentences, and vowels
2. A group box called, "grpActions" must be created and must contain the buttons for: process, clear, and exit

     vii.   File Contents Textbox
1. The name of this textbox must be "txtFileContents"
2. The ReadOnly property for this textbox must be set to True
3. The Multiline property for this textbox must be set to True
4. The Scroolbars property for this textbox must be set to Vertical

    viii.   Character Count Textbox
1. The name of this textbox must be "txtCharacterCount"
2. The ReadOnly property for this textbox must be set to True

     ix.   Word Count Textbox
1. The name of this textbox must be "txtWordCount"
2. The ReadOnly property for this textbox must be set to True

      x.   Sentence Count Textbox
1. The name of this textbox must be "txtSentenceCount"
2. The ReadOnly property for this textbox must be set to True

     xi.   Vowel Count Textbox
1. The name of this textbox must be "txtVowelCount"
2. The ReadOnly property for this textbox must be set to True

     xii.   Labels must be created as shown in Figure 1
1. All labels must be prefixed with "lbl". For example, lblCharacters is the label name for the label with the text "Characters" in Figure 1

3. **Tasks (see Table 1 below)**
   a. Close the Application Button Click
      i. Closes the form. Upon closing, the application must confirm that the form is to be closed and allow the user to cancel the close by responding with "No" to the question, "Do you wish to exit this application?" If the user responds with a "Yes", then the application is closed.
     ii. The dialog box used to solicit the response must contain the message, "Do you wish to exit this application?" with a caption of "Exit?"
    iii. The dialog box used to solicit the response must contain two buttons: Yes and No.

    b. Clear the Form Button Click
        i. Clears all output fields by setting them to the value vbNullString
        ii. Fields to set to vbNullString: txtFileName, txtCharacterCount, txtWordCount, txtSentenceCount, and txtVowelCount.
        iii. Disable the clear button
        iv. Disable the process button
    c. Process a File Button Click
        i. Opens the file contained in txtFileName for input
        ii. Read the contents of the file into the txtFileContents control for on-screen display
        iii. Count the characters in the file and display the count in txtCharacterCount
        iv. Count the words in the file and display the count in txtWordCount.
        v. Count the sentences in the file and display the count in txtSentenceCount
        vi. Count the vowels in the file and display the count in txtVowelCount.
        vii. Enable the clear button
        viii. Enable the process button
    d. Choose a File Button Click
        i. Uses the Open File Dialog object to browse for a file to read
        ii. The Open File Dialog object's filter property must be set to "Text files (*.txt)|*.txt" or "*.txt|*.txt"

Task	Object	Event
Close the Application	Exit Button	Click
Clear the Form	Clear Button	Click
Process a File	Process Button	Click
Choose a File	Choose File Button	Click

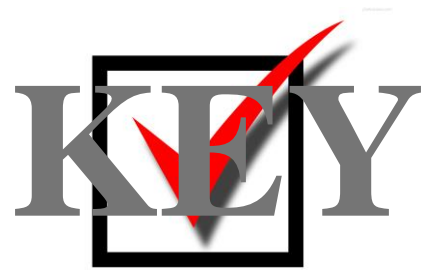Table 1 - Task-Object-Event Chart

Development Standards

- Standard name prefixes must be utilized for forms, controls, and variables.
- All subroutines, functions, and methods must be documented with comments explaining the purpose of the method, the input parameters (if any), and the output (if any).
- The user interface must be similar to Figure 1.

User Interface Design

Your application will be graded on the following criteria:

Solution and Project

The project is present on the flash drive ____ 10 pts
The project is named according to the naming conventions ____ 10 pts

User Interface - Property Checks

The user interface design resembles Figure 1 ____ 20 pts
The name of the form is, "frmMain" ____ 5 pts
The form's caption displays "Submission Statistics: Contestant #ABC",
where#ABC is the contestant's properly formatted BPA ID ____ 5 pts
The user interface's Start Position is set to "CenterScreen" ____ 5 pts

The user interface contains a button named, "cmdExit" ____ 5 pts
The user interface contains a button captioned, "E&xit" ____ 5 pts

The user interface contains a button named, "cmdClear" ____ 5 pts
The user interface contains a button captioned, "Clear" ____ 5 pts
The enabled property for this button must be set to False ____ 5 pts

The user interface contains a button named, "cmdProcess" ____ 5 pts
The user interface contains a button captioned, "Process" ____ 5 pts
The enabled property for this button must be set to False ____ 5 pts

The user interface contains a button named, "cmdChooseFile" ____ 5 pts
The user interface contains a button captioned, "Choose File" ____ 5 pts

The user interface contains a textbox called, "txtFileName" ____ 5 pts
The "txtFileName" textbox's ReadOnly property is set to True ____ 5 pts

The user interface contains a textbox called, "txtFileContents" ____ 5 pts
The "txtFileContents" textbox's ReadOnly property is set to True ____ 5 pts
The "txtFileContents" textbox's Multiline property is set to True ____ 5 pts
The "txtFileContents" textbox's Scrollbars property is set to Vertical ____ 5 pts

The user interface contains a textbox called, "txtCharacterCount" ____ 5 pts
The "txtCharacterCount" textbox's ReadOnly property is set to True ____ 5 pts

The user interface contains a textbox called, "txtWordCount" ____ 5 pts
The "txtWordCount" textbox's ReadOnly property is set to True ____ 5 pts

The user interface contains a textbox called, "txtSentenceCount" ____ 5 pts
The "txtSentenceCount" textbox's ReadOnly property is set to True ____ 5 pts

The user interface contains a textbox called, "txtVowelCount" ＿＿＿ 5 pts
The "txtVowelCount" textbox's ReadOnly property is set to True ＿＿＿ 5 pts

The following labels are present: "File Name", "Characters",
"Words", "Sentences", and Vowels" (2 point each) ＿＿＿ 10 pts

Runtime Checks

Code copied to USB drive and program runs from the USB drive ＿＿＿ 10 pts

If the program does not execute, then the remaining items in this section receive a score of zero

When pressed, the cmdClear button clears the contents of the
txtCharacterCount, txtWordCount, txtSentenceCount, txtVowelCount,
and txtFileContents textboxes (2 point each) ＿＿＿ 10 pts

When pressed, the cmdProcess button processes the selected file
and populates the txtCharacterCount, txtWordCount, txtSentenceCount,
txtVowelCount, and txtFileContens textboxes (20 points each) ＿＿＿ 100 pts

After the cmdProcess button is pressed, the cmdClear button is
Enabled ＿＿＿ 10 pts

When pressed, the cmdClear button clears both the txtEnglish and the
txtMorseCode textboxes (5 points each) ＿＿＿ 10 pts

When pressed, the cmdChooseFile button displays an OpenFileDialog
Screen and allows the user to select a file from the list ＿＿＿ 20 pts

The open file dialog filter is set to "Text files (*.txt)|*.txt" or "*.txt|*.txt" ＿＿＿ 15 pts

When pressed, the cmdExit button attempts to close the form ＿＿＿ 10 pts

When the user attempts to close the application, a confirmation dialog
is presented that confirms that the user wishes to exit the
application ＿＿＿ 20 pts
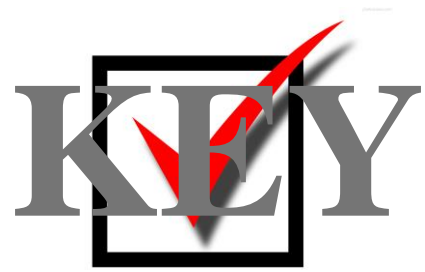
Source Code Review

Code is commented at the top, for each function, and as needed	____ 15 pts
Code uses a consistent variable naming conventions	____ 15 pts
Code is present to open the input file	____ 20 pts
Code is present to read from the input file	____ 10 pts
Error handling is used to catch and handle errors when opening the file	____ 40 pts
The input file is closed	____ 10 pts

Total Points: 500 pts

Solution Screen Shot
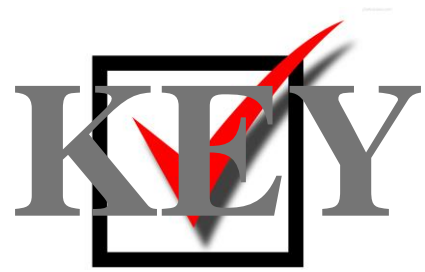
Sample Code

```vb
Public Class frmMain
    Private Sub cmdChooseFile_Click(sender As System.Object, e As System.EventArgs) Handles cmdChooseFile.Click
        Dim ofd As OpenFileDialog = New OpenFileDialog()

        ofd.Filter = "Text files (*.txt)|*.txt"
        ofd.ShowDialog()

        If (Not ofd.FileName.Length = 0) Then
            txtFileName.Text = ofd.FileName
            cmdProcess.Enabled = True
        End If
    End Sub

    Private Sub cmdExit_Click(sender As System.Object, e As System.EventArgs) Handles cmdExit.Click
        Close()
    End Sub

    Private Sub cmdProcess_Click(sender As System.Object, e As System.EventArgs) Handles cmdProcess.Click
        Dim sr As System.IO.StreamReader = New System.IO.StreamReader(txtFileName.Text)
        txtFileName.Text = sr.ReadToEnd()
        sr.Close()

        txtFileContents.Text = txtFileName.Text
        txtCharacterCount.Text = CountCharacters(txtFileName.Text)
        txtWordCount.Text = CountWords(txtFileName.Text)
        txtSentenceCount.Text = CountSentences(txtFileName.Text)
        txtVowelCount.Text = CountVowels(txtFileName.Text)
        cmdClear.Enabled = True
        cmdProcess.Enabled = False
    End Sub

    Private Function CountCharacters(ByRef text As String) As Integer
        Dim count As Integer = 0
        count = text.Length
        Return count
    End Function

    Private Function CountWords(ByRef text As String) As Integer
        Dim words As String() = text.Split(" ")
        Return words.Length
    End Function

    Private Function CountSentences(ByRef text As String) As Integer
        Dim words As String() = text.Split(".!?")
        Return words.Length - 1
    End Function

    Private Function CountVowels(ByRef text As String) As Integer
        Dim count As Integer = 0
        For Each ch As Char In text
            If isVowel(ch) Then
                count = count + 1
```
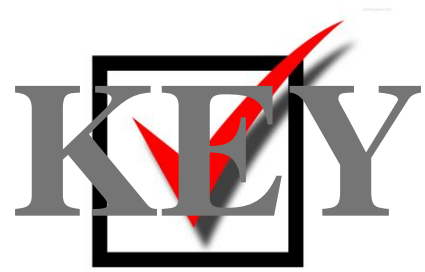
```vb
            End If
        Next
        Return count
    End Function

    Private Function isVowel(ByVal ch As Char) As Boolean
        Dim vowels As String = "AEIOU"
        ch = Char.ToUpper(ch)
        Return vowels.IndexOf(ch) <> -1
    End Function

    Private Sub frmMain_FormClosing(sender As System.Object, e As
System.Windows.Forms.FormClosingEventArgs) Handles MyBase.FormClosing
        Dim result As DialogResult = MessageBox.Show("Do you wish to exit this
application?", "Exit?", MessageBoxButtons.YesNo)
        If (result = Windows.Forms.DialogResult.Yes) Then
            e.Cancel = False
        Else
            e.Cancel = True
        End If
    End Sub

    Private Sub cmdClear_Click(sender As System.Object, e As System.EventArgs) Handles
cmdClear.Click
        txtFileName.Text = vbNullString
        txtFileName.Text = vbNullString
        txtCharacterCount.Text = vbNullString
        txtSentenceCount.Text = vbNullString
        txtVowelCount.Text = vbNullString
        txtWordCount.Text = vbNullString
        cmdClear.Enabled = False
        cmdProcess.Enabled = False
    End Sub
End Class
```