Question 1:

```vb
Imports System.IO
Public Class Form1    'Ilker Hadzhalaran

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        rad1DArray.Checked = True
        lblDisplayResults.Text = String.Empty
        Dim outFile As StreamWriter

        outFile = File.CreateText("C:\Users\Ilker\Desktop\Data.txt")

        outFile.WriteLine("10    5")
        outFile.WriteLine("15    20")
        outFile.WriteLine("20    18")
        outFile.WriteLine("25    40")
        outFile.WriteLine("40    33")
        outFile.WriteLine("50    54")
        outFile.WriteLine("55    70")
        outFile.WriteLine("60    60")
        outFile.WriteLine("75    78")

        outFile.Close()

    End Sub

    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
        rad1DArray.Checked = True
        lblDisplayResults.Text = String.Empty
    End Sub

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub

    Private Sub btnEstimate_Click(sender As Object, e As EventArgs) Handles
btnEstimate.Click

        Dim dblCalculateM, dblCalculateB, dblCalculateAverageT, dblCalculateAverageS,
dblCalculateS As Double
        Dim dblTimeEntered As Double
        Const intNUMBER_TERMS As Integer = 9

        lblDisplayResults.Text = String.Empty

        dblTimeEntered = getTime()

        If dblTimeEntered < 10 Or dblTimeEntered > 75 Then
            MsgBox("Time was either less than 10min or more than 75min")
            Exit Sub
        End If


        If rad1DArray.Checked = True Then

            Dim dblTime() As Double = {10, 15, 20, 25, 40, 50, 55, 60, 75}
            Dim dblStrength() As Double = {5, 20, 18, 40, 33, 54, 70, 60, 78}
```

```vb
                Tensile_1D(dblTimeEntered, intNUMBER_TERMS, dblTime, dblStrength,
dblCalculateM, dblCalculateB, dblCalculateAverageT, dblCalculateAverageS, dblCalculateS)

                printValues(dblCalculateM, dblCalculateB, dblTimeEntered, dblCalculateS)

        ElseIf rad2DArray.Checked = True Then

                Dim dblTimeAndStrength(,) As Double = {{10, 15, 20, 25, 40, 50, 55, 60, 75},
{5, 20, 18, 40, 33, 54, 70, 60, 78}}

                Tensile_2D(dblTimeEntered, intNUMBER_TERMS, dblTimeAndStrength,
dblCalculateM, dblCalculateB, dblCalculateAverageT, dblCalculateAverageS, dblCalculateS)

                printValues(dblCalculateM, dblCalculateB, dblTimeEntered, dblCalculateS)

        ElseIf radFunction.Checked = True Then

                Const intARRAY_ROWS As Integer = 2
                Const intARRAY_COLUMNS As Integer = 9
                Dim dblTimeAndStrength(intARRAY_ROWS - 1, intARRAY_COLUMNS - 1) As Double

                userEnteredArray(dblTimeAndStrength, intARRAY_ROWS, intARRAY_COLUMNS)

                Tensile_2D(dblTimeEntered, intNUMBER_TERMS, dblTimeAndStrength,
dblCalculateM, dblCalculateB, dblCalculateAverageT, dblCalculateAverageS, dblCalculateS)

                printValues(dblCalculateM, dblCalculateB, dblTimeEntered, dblCalculateS)

        ElseIf radTextFile.Checked = True Then

                If Not File.Exists("C:\Users\Ilker\Desktop\Data.txt") Then
                    MsgBox("Data file does not exist.")
                    Exit Sub
                End If

                Dim dblTimeAndStrength(1, 8) As Double
                Dim inputFile As StreamReader
                Dim dblTimeDataPoint, dblStrengthDataPoint As Double
                Dim strDataReader(1) As String
                Dim intCounter As Integer = -1
                inputFile = File.OpenText("C:\Users\Ilker\Desktop\Data.txt")

                Do Until inputFile.EndOfStream

                    strDataReader = Split(inputFile.ReadLine(), "   ")

                    intCounter += 1
                    dblTimeAndStrength(0, intCounter) = CDbl(strDataReader(0))
                    dblTimeAndStrength(1, intCounter) = CDbl(strDataReader(1))

                Loop

                inputFile.Close()

                Tensile_2D(dblTimeEntered, intNUMBER_TERMS, dblTimeAndStrength,
dblCalculateM, dblCalculateB, dblCalculateAverageT, dblCalculateAverageS, dblCalculateS)

                printValues(dblCalculateM, dblCalculateB, dblTimeEntered, dblCalculateS)
```

```vbnet
        End If

    End Sub

    Private Function getTime() As Double

        Dim dblTimeEntered As Double

        dblTimeEntered = CInt(InputBox("Enter any time between 10-75min"))

        Return dblTimeEntered


    End Function

    Private Sub Tensile_1D(ByVal dblTimeEntered As Double, ByVal intNUMBER_TERMS As
Integer, ByVal dblTime() As Double, ByVal dblStrength() As Double, ByRef dblCalculateM As
Double, ByRef dblCalculateB As Double, ByRef dblCalculateAverageT As Double, ByRef
dblCalculateAverageS As Double, ByRef dblCalculateS As Double)

        Dim dblTermTS, dblTermT, dblTermS, dblTermTSquared As Double

        For intCounter As Integer = 0 To intNUMBER_TERMS - 1 Step 1

            dblTermTS += dblTime(intCounter) * dblStrength(intCounter)
            dblTermT += dblTime(intCounter)
            dblTermS += dblStrength(intCounter)
            dblTermTSquared += (dblTime(intCounter)) ^ 2

        Next intCounter

        dblCalculateM = (intNUMBER_TERMS * dblTermTS - dblTermT * dblTermS) /
(intNUMBER_TERMS * dblTermTSquared - dblTermT ^ 2)
        dblCalculateAverageT = dblTermT / intNUMBER_TERMS
        dblCalculateAverageS = dblTermS / intNUMBER_TERMS
        dblCalculateB = dblCalculateAverageS - dblCalculateM * dblCalculateAverageT
        dblCalculateS = dblCalculateM * dblTimeEntered + dblCalculateB

    End Sub

    Private Sub Tensile_2D(ByVal dblTimeEntered As Double, ByVal intNUMBER_TERMS As
Integer, ByVal dblTimeAndStrength(,) As Double, ByRef dblCalculateM As Double, ByRef
dblCalculateB As Double, ByRef dblCalculateAverageT As Double, ByRef dblCalculateAverageS
As Double, ByRef dblCalculateS As Double)

        Dim dblTermTS, dblTermT, dblTermS, dblTermTSquared As Double

        For intCounter As Integer = 0 To intNUMBER_TERMS - 1 Step 1

            dblTermTS += dblTimeAndStrength(0, intCounter) * dblTimeAndStrength(1,
intCounter)
            dblTermT += dblTimeAndStrength(0, intCounter)
            dblTermS += dblTimeAndStrength(1, intCounter)
            dblTermTSquared += dblTimeAndStrength(0, intCounter) ^ 2
```

```vbnet
            Next intCounter

            dblCalculateM = (intNUMBER_TERMS * dblTermTS - dblTermT * dblTermS) /
    (intNUMBER_TERMS * dblTermTSquared - dblTermT ^ 2)
            dblCalculateAverageT = dblTermT / intNUMBER_TERMS
            dblCalculateAverageS = dblTermS / intNUMBER_TERMS
            dblCalculateB = dblCalculateAverageS - dblCalculateM * dblCalculateAverageT
            dblCalculateS = dblCalculateM * dblTimeEntered + dblCalculateB

        End Sub

        Private Function userEnteredArray(ByRef dblTimeAndStrength(,) As Double, ByVal
    intARRAY_ROWS As Integer, ByVal intARRAY_COLUMNS As Integer) As Array

            For intCounter As Integer = 0 To intARRAY_ROWS - 1 Step 1

                If intCounter = 0 Then

                    For intCounter2 As Integer = 0 To intARRAY_COLUMNS - 1 Step 1
                        dblTimeAndStrength(intCounter, intCounter2) = CDbl(InputBox("Enter
    time for datapoint #" & CStr(intCounter2 + 1)))

                    Next intCounter2

                ElseIf intCounter = 1 Then

                    For intCounter2 As Integer = 0 To intARRAY_COLUMNS - 1 Step 1
                        dblTimeAndStrength(intCounter, intCounter2) = CDbl(InputBox("Enter
    tensile strength for datapoint #" & CStr(intCounter2 + 1)))

                    Next intCounter2

                End If

            Next intCounter

            Return dblTimeAndStrength


        End Function


        Private Sub printValues(ByVal dblCalculateM As Double, ByVal dblCalculateB As Double,
    ByVal dblTimeEntered As Double, ByVal dblCalculateS As Double)
            lblDisplayResults.Text &= "m = " & dblCalculateM.ToString("f3") & ", b = " &
    dblCalculateB.ToString("f3") & vbCr
            lblDisplayResults.Text &= "t = " & dblTimeEntered.ToString("f1") & " min, " &
    "Estimated Tensile Strength = " & dblCalculateS.ToString("f1") & " s"
        End Sub


    End Class
```

Form1

Pick Method:
- ( ) 1D Array   ( ) 2D Array   ( ) Function   ( ) Text File

[Estimate]   [Clear]   [Exit]

Assignment3_Question1

Enter any time between 10-75min

[OK]

[Cancel]

`11`

Form1

Pick Method:
- ( ) 1D Array   ( ) 2D Array   ( ) Function   ( ) Text File

m = 1.059, b = 0.818
t = 11.0 min, Estimated Tensile Strength = 12.5 s

[Estimate]   [Clear]   [Exit]

Form1

Pick Method:
- ○ 1D Array ● 2D Array ○ Function ○ Text File

[Estimate]   [Clear]   [Exit]

Assignment3_Question1

Enter any time between 10-75min

[OK]
[Cancel]

11

Form1

Pick Method:
- ○ 1D Array ● 2D Array ○ Function ○ Text File

m = 1.059, b = 0.818
t = 11.0 min, Estimated Tensile Strength = 12.5 s

[Estimate]   [Clear]   [Exit]

Form1

Pick Method:
- ( ) 1D Array
- ( ) 2D Array
- (•) Function
- ( ) Text File

[Estimate] [Clear] [Exit]

Assignment3_Question1

Enter any time between 10-75min

[OK]
[Cancel]

```
11
```

Form1

Pick Method:
- ( ) 1D Array
- ( ) 2D Array
- (•) Function
- ( ) Text File

[Estimate] [Clear] [Exit]

Assignment3_Question1

Enter time for datapoint #1

[OK]
[Cancel]

```
10
```

Form1

Pick Method:
- ○ 1D Array ○ 2D Array ⦿ Function ○ Text File

[Estimate]   [Clear]   [Exit]

Assignment3_Question1

Enter time for datapoint #9

[OK]

[Cancel]

75

Form1

Pick Method:
- ○ 1D Array ○ 2D Array ⦿ Function ○ Text File

[Estimate]   [Clear]   [Exit]

Assignment3_Question1

Enter tensile strength for datapoint #1

[OK]

[Cancel]

5

Form1

Pick Method:
- ○ 1D Array
- ○ 2D Array
- ● Function
- ○ Text File

m = 1.059, b = 0.818
t = 11.0 min, Estimated Tensile Strength = 12.5 s

[Estimate]   [Clear]   [Exit]

Form1

Pick Method:
- ○ 1D Array
- ○ 2D Array
- ○ Function
- ● Text File

[Estimate]   [Clear]   [Exit]

Assignment3_Question1

Enter any time between 10-75min

[OK]
[Cancel]

11

Form1

Pick Method:

( ) 1D Array   ( ) 2D Array   ( ) Function   (•) Text File

m = 1.059, b = 0.818
t = 11.0 min, Estimated Tensile Strength = 12.5 s

[Estimate]   [Clear]   [Exit]

Data - Notepad

File Edit Format View Help

```
10    5
15    20
20    18
25    40
40    33
50    54
55    70
60    60
75    78
```

Question 2:

```vbnet
Imports System.IO
Public Class Form1    'Ilker Hadzhalaran

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load

    End Sub

    Private Sub btnFindValue_Click(sender As Object, e As EventArgs) Handles
btnFindValue.Click

        Dim inputFile As StreamReader
        Dim outFile As StreamWriter
        Dim strFileReader(2) As String
        Dim dblPressure As Double = 0
        Dim dblTemperature As Double = 0
        Dim dblVolume As Double = 0
        Dim strPropertyName As String
        Const dblMOLE_NUMBER As Double = 1000
        Const dblR_CONSTANT As Double = 8.314

        inputFile = File.OpenText("C:\Users\Ilker\Desktop\idealGas.txt")

        If Not File.Exists("C:\Users\Ilker\Desktop\idealGas.txt") Then

            MsgBox("File does not exist.")
            Exit Sub
        End If


        Do Until inputFile.EndOfStream
            strFileReader = Split(inputFile.ReadLine(), " ")
            strPropertyName = strFileReader(0)

            If strPropertyName = "Pressure" Then
                dblPressure = CDbl(strFileReader(1))

            ElseIf strPropertyName = "Temperature" Then
                dblTemperature = CDbl(strFileReader(1))
            ElseIf strPropertyName = "Volume" Then
                dblVolume = CDbl(strFileReader(1))

            End If


        Loop

        inputFile.Close()

        outFile = File.AppendText("C:\Users\Ilker\Desktop\idealGas.txt")

        If dblPressure = 0 Then
            dblPressure = dblMOLE_NUMBER * dblR_CONSTANT * dblTemperature / dblVolume
```

```vbnet
            lblDisplayResults.Text &= "Unknown Pressure is: " &
dblPressure.ToString("f2") & " Pa" & vbCr

            lblDisplayResults.Text &= "R constant is: " & dblR_CONSTANT.ToString("f2") &
" Jk^-1*mol^-1" & vbCr
            lblDisplayResults.Text &= "Number of Moles is: " &
dblMOLE_NUMBER.ToString("f2") & " mol" & vbCr
            lblDisplayResults.Text &= "Temperature is: " & dblTemperature.ToString("f2")
& " K" & vbCr
            lblDisplayResults.Text &= "Volume is: " & dblVolume.ToString("f2") & " m^3" &
vbCr

            outFile.WriteLine("")
            outFile.WriteLine("Pressure " & dblPressure.ToString("f2") & " Pa")

        ElseIf dblTemperature = 0 Then
            dblTemperature = dblPressure * dblVolume / dblMOLE_NUMBER / dblR_CONSTANT

            lblDisplayResults.Text &= "Unknown Temperature is: " &
dblTemperature.ToString("f2") & " K" & vbCr

            lblDisplayResults.Text &= "R constant is: " & dblR_CONSTANT.ToString("f2") &
" Jk^-1*mol^-1" & vbCr
            lblDisplayResults.Text &= "Number of Moles is: " &
dblMOLE_NUMBER.ToString("f2") & " mol" & vbCr
            lblDisplayResults.Text &= "Pressure is: " & dblPressure.ToString("f2") & "
Pa" & vbCr
            lblDisplayResults.Text &= "Volume is: " & dblVolume.ToString("f2") & " m^3" &
vbCr

            outFile.WriteLine("")
            outFile.WriteLine("Temperature " & dblTemperature.ToString("f2") & " K")

        ElseIf dblVolume = 0 Then
            dblVolume = dblMOLE_NUMBER * dblR_CONSTANT * dblTemperature / dblPressure

            lblDisplayResults.Text &= "Unknown Volume is: " & dblVolume.ToString("f2") &
" m^3" & vbCr

            lblDisplayResults.Text &= "R constant is: " & dblR_CONSTANT.ToString("f2") &
" Jk^-1*mol^-1" & vbCr
            lblDisplayResults.Text &= "Number of Moles is: " &
dblMOLE_NUMBER.ToString("f2") & " mol" & vbCr
            lblDisplayResults.Text &= "Temperature is: " & dblTemperature.ToString("f2")
& " K" & vbCr
            lblDisplayResults.Text &= "Pressure is: " & dblPressure.ToString("f2") & "
Pa" & vbCr

            outFile.WriteLine("")
            outFile.WriteLine("Volume " & dblVolume.ToString("f2") & " m^3")

        End If

        outFile.Close()
    End Sub


    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
```

```vbnet
            lblDisplayResults.Text = String.Empty

    End Sub

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub
End Class
```
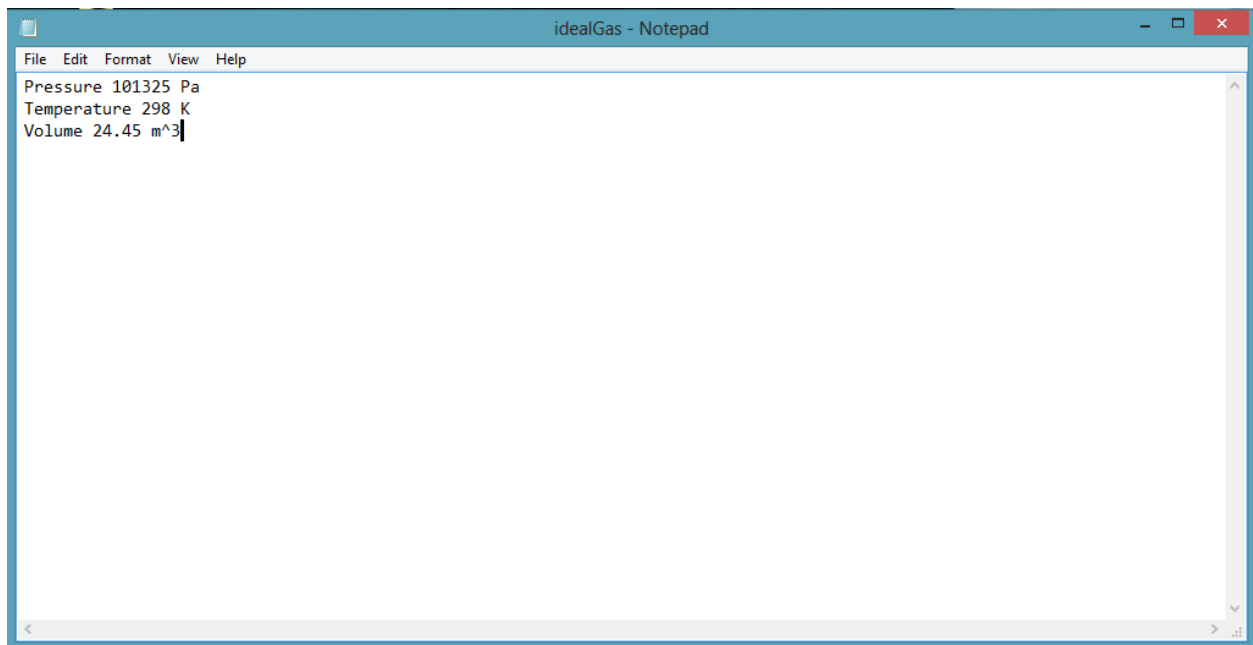
idealGas - Notepad

File   Edit   Format   View   Help

```
Pressure 101325 Pa
Temperature 298 K
```

Form1

Unknown Volume is: 24.45 m^3
R constant is: 8.31 Jk^-1*mol^-1
Number of Moles is: 1000.00 mol
Temperature is: 298.00 K
Pressure is: 101325.00 Pa

[Find Value]   [Clear]   [Exit]

idealGas - Notepad

File Edit Format View Help

```
Pressure 101325 Pa
Temperature 298 K
Volume 24.45 m^3
```

Question 3:

```vb
Public Class Form1    'Ilker Hadzhalaran

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub

    Private Sub btnClick_Click(sender As Object, e As EventArgs) Handles btnClick.Click


        Dim g As Graphics = picDrawing.CreateGraphics()
        Dim Pen As New Pen(Color.Yellow)
        Dim Brush As New SolidBrush(Color.Yellow)

        Static intClickCounter = 0
        intClickCounter += 1
        btnClick.Text = "Keep clicking..."


        Select Case intClickCounter
            Case 1
                Pen.Color = Color.LightYellow
                g.DrawEllipse(Pen, 90, 90, 90, 90)
                Brush.Color = Color.Yellow
                g.FillEllipse(Brush, 90, 90, 90, 90)

            Case 2

                Pen.Color = Color.Brown
                g.DrawRectangle(Pen, 70, 300, 90, 90)
                Brush.Color = Color.Brown
                g.FillRectangle(Brush, 70, 300, 90, 90)

            Case 3

                Brush.Color = Color.Red
                Dim A As New Point(115, 250)
                Dim B As New Point(68, 300)
                Dim C As New Point(162, 300)
                Dim triangle As Point() = {A, B, C}
                Pen.Width = 4
                Pen.Color = Color.Red
                g.DrawPolygon(Pen, triangle)
                Brush.Color = Color.Red
                g.FillPolygon(Brush, triangle)

            Case 4

                Brush.Color = Color.Cyan
                g.FillRectangle(Brush, 120, 330, 20, 20)

            Case 5
                Brush.Color = Color.Salmon
                g.FillRectangle(Brush, 80, 351, 20, 40)
```

```vbnet
                Case 6
                    Brush.Color = Color.Brown
                    g.FillRectangle(Brush, 305, 270, 30, 120)


                Case 7

                    Pen.Color = Color.Green

                    For intTreeLeaves As Integer = 250 To 300 Step 1
                        g.DrawArc(Pen, intTreeLeaves, 270, 90, 90, 180, 180)

                    Next intTreeLeaves

            End Select




    End Sub

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub
End Class
```

Form1

Click to show the drawing:

Exit