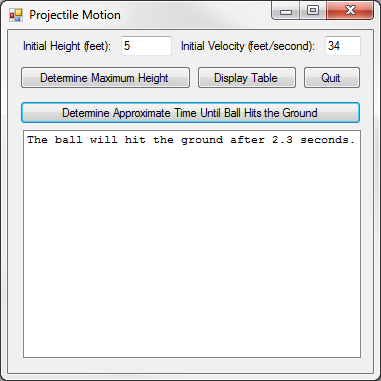
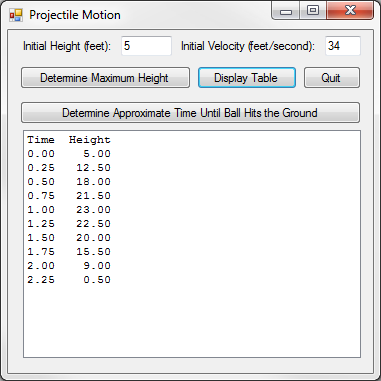
3:





Public Class frm3

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

lstOutput.Items.Clear()

Dim h As Decimal = CDec(txtHeight.Text)

Dim v As Decimal = CDec(txtVelocity.Text)

Dim t As Decimal = v / 32

Dim MaxHeight As Decimal = h + (v \* t) - (16 \* (t ^ 2))

lstOutput.Items.Add("The maximum height is " & MaxHeight & " feet.")

End Sub

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

lstOutput.Items.Clear()

Dim h As Decimal = CDec(txtHeight.Text)

Dim v As Decimal = CDec(txtVelocity.Text)

Dim t As Decimal = 0

Dim Height As Decimal = h + (v \* t) - (16 \* (t ^ 2))

Do While Height > 0

t += 0.1

Height = h + (v \* t) - (16 \* (t ^ 2))

Loop

lstOutput.Items.Add("The ball will hit the ground after " & t & " seconds.")

End Sub

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

lstOutput.Items.Clear()

Dim f As String = "{0,-6:n2}{1,6:n2}"

Dim h As Decimal = CDec(txtHeight.Text)

Dim v As Decimal = CDec(txtVelocity.Text)

Dim Height As Decimal = 0

lstOutput.Items.Add(String.Format(f, "Time", "Height"))

For t As Double = 0 To 5 Step 0.25

Height = h + (v \* t) - (16 \* (t ^ 2))

If Height < 0 Then Exit Sub

lstOutput.Items.Add(String.Format(f, t, Height))

Next

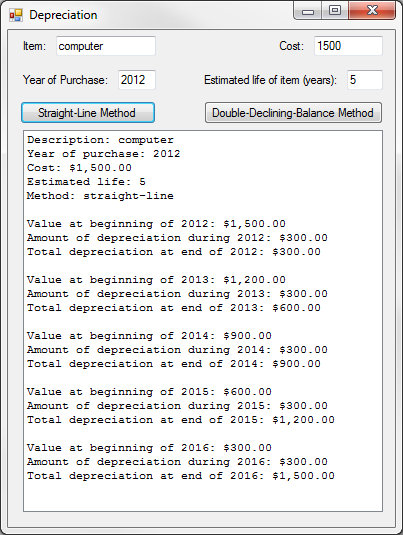
End Sub

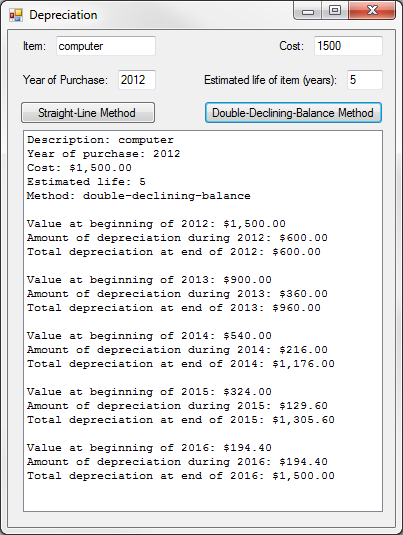
Private Sub btnQuit\_Click(sender As Object, e As EventArgs) Handles btnQuit.Click

End

End Sub

End Class

4:



Public Class frm4

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

lstOutput.Items.Clear()

lstOutput.Items.Add("Description: " & txtItem.Text)

Dim Year As Decimal = CDec(txtYear.Text)

lstOutput.Items.Add("Year of purchase: " & Year)

Dim Cost As Decimal = CDec(txtCost.Text)

lstOutput.Items.Add("Cost: " & FormatCurrency(Cost))

Dim Years As Decimal = CDec(txtYears.Text)

lstOutput.Items.Add("Estimated life: " & Years)

lstOutput.Items.Add("Method: straight-line")

Dim Dep As Decimal = (1 / Years) \* Cost

Dim TDep As Decimal = 0

Years = Year + Years - 1

For i As Integer = Year To Years Step 1

lstOutput.Items.Add("")

lstOutput.Items.Add("Value at beginning of " & i & ": " & FormatCurrency(Cost))

Cost -= Dep

lstOutput.Items.Add("Amount of depreciation during " & i & ": " & FormatCurrency(Dep))

TDep += Dep

lstOutput.Items.Add("Total depreciation at end of " & i & ": " & FormatCurrency(TDep))

Next

End Sub

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

lstOutput.Items.Clear()

lstOutput.Items.Add("Description: " & txtItem.Text)

Dim Year As Decimal = CDec(txtYear.Text)

lstOutput.Items.Add("Year of purchase: " & Year)

Dim Cost As Decimal = CDec(txtCost.Text)

lstOutput.Items.Add("Cost: " & FormatCurrency(Cost))

Dim Years As Decimal = CDec(txtYears.Text)

lstOutput.Items.Add("Estimated life: " & Years)

lstOutput.Items.Add("Method: double-declining-balance")

Dim Dep As Decimal = 0

Dim TDep As Decimal = 0

Dim TYears = Year + Years - 1

For i As Integer = Year To TYears Step 1

lstOutput.Items.Add("")

Dep = (2 / Years) \* Cost

If i = TYears Then Dep = Cost

lstOutput.Items.Add("Value at beginning of " & i & ": " & FormatCurrency(Cost))

Cost -= Dep

lstOutput.Items.Add("Amount of depreciation during " & i & ": " & FormatCurrency(Dep))

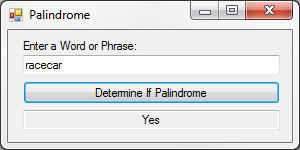
TDep += Dep

lstOutput.Items.Add("Total depreciation at end of " & i & ": " & FormatCurrency(TDep))

Next

End Sub

End Class

7:

Public Class frm7

Function IsPalindrome(Input As String) As Boolean

Dim CleanInput As String = ""

Dim NextChar As String = ""

For i As Integer = 0 To Input.Length - 1

NextChar = Input.Substring(i, 1).ToLower

If NextChar Like "[a-z]" Then CleanInput += NextChar

Next

Dim InputRev As String = StrReverse(CleanInput).ToLower

If CleanInput = InputRev Then Return True

Return False

End Function

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

If IsPalindrome(txtInput.Text) Then txtOutput.Text = "Yes" Else txtOutput.Text = "No"

End Sub

End Class