Visual Basic .Net

## Questions: 17/17 48/50

Program: 31/33

* When I add Option Strict On, I do get an error on the following statement:

intLaborValue = intLaborValue \* intLABOR\_PER\_HOUR

You are multiplying an integer by a decimal and storing it into an integer. Change it to:

intLaborValue = CInt(intLaborValue \* intLABOR\_PER\_HOUR)

* If negative numbers have been entered, you wouldn’t want to proceed to do any calculations. If you had checked for the negatives in your calculate click event handler, you could have just skipped over invoking the other functions (and doing calculations) if a negative had ben entered. It would be the simpliest approach to implement; also it “relieves” the functions from having to do any GUI validations – their purpose will be calculations only. I kinda went overboard with trying the changes in your code. Take a look at what I have here, and hopefully it will make sense. I pushed the validation all into the calculate and it seems to work ok.

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

Dim intLaborValue As Integer = 0

Dim decPartsValue As Decimal = 0D

Dim decServiceLaborValue As Decimal = 0D

Dim boolError As Boolean = False

' Validationg Labor input box

If txtLabor.Text = String.Empty Then

' if there isn't any labor charges

decServiceLaborValue = OilLubeCharges() + FlushCharges() + MiscCharges()

Else

If Integer.TryParse(txtLabor.Text, intLaborValue) Then

If intLaborValue >= 0 Then

intLaborValue = CInt(intLaborValue \* intLABOR\_PER\_HOUR)

' if there are labor charges

decServiceLaborValue = Convert.ToDecimal(intLaborValue) + OilLubeCharges() + FlushCharges() + MiscCharges()

Else

boolError = True

MessageBox.Show("Must enter positive value for hours")

End If

Else

MessageBox.Show("Hours worked must be billed as full hours.")

boolError = True

End If

End If

If Not boolError Then

' ----- Display values into the Summary Labels -----

'lblServicesLabor.Text = decServiceLaborValue.ToString("c") ' Service & Labor Label

' Validating Parts input box ' Parts & Tax Label(tax if parts is not empty)

If txtParts.Text = String.Empty Then

lblParts.Text = decPartsValue.ToString("c")

lblTax.Text = "$0.00"

Else

If Decimal.TryParse(txtParts.Text, decPartsValue) Then

If decPartsValue >= 0 Then

lblParts.Text = decPartsValue.ToString("c")

lblTax.Text = TaxCharges(decPartsValue).ToString("c")

Else

MessageBox.Show("Value entered for parts must be 0 or more")

boolError = True

End If

Else

MessageBox.Show("Value entered for parts must be numeric")

boolError = True

End If

End If

End If

If Not boolError Then

lblServicesLabor.Text = decServiceLaborValue.ToString("c") ' Service & Labor Label

lblTotalFees.Text = TotalCharges(decPartsValue, decServiceLaborValue).ToString("c") ' Total Fees Label

' Services Perform Summary List to Customer

lblServicesPerformed.Visible = True

lblSummaryList.Text = DisplaySummaryList(decPartsValue, intLaborValue)

lblSummaryList.Visible = True

End If

End Sub

Function OtherCharges(ByVal decPartsValue As Decimal, ByVal intLaborValue As Integer) As Decimal

Dim decOtherCharges As Decimal = 0D

'Dim decPartsValue As Decimal = 0D

' Dim intLaborValue As Integer = 0

' Values have to be positive numbers & have numbers

' If txtParts.Text <> String.Empty Then

'If Decimal.TryParse(txtParts.Text, decPartsValue) Then

'If decPartsValue >= 0 Then

decOtherCharges += decPartsValue

'Else

' MessageBox.Show("Parts charge cannot be negative amount.")

' End If

' Else

' MessageBox.Show("Please enter a numeric value into Parts.")

'End If

'End If

' Values have to be positive numbers & have numbers

' If txtLabor.Text <> String.Empty Then

'If Integer.TryParse(txtLabor.Text, intLaborValue) Then

'If intLaborValue > 0 Then

decOtherCharges += (intLaborValue \* intLABOR\_PER\_HOUR)

'Else

'MessageBox.Show("Labor charge cannot be negative amount.")

' End If

'Else

'MessageBox.Show("Please enter a numeric value into Labor.")

'End If

' End If

Return decOtherCharges

End Function

Function TaxCharges(ByVal decPartsValue As Decimal) As Decimal

Dim decTaxOnParts As Decimal = 0D

' Dim decPartsValue As Decimal = 0D

' Tax on parts only

'If Decimal.TryParse(txtParts.Text, decPartsValue) Then

decTaxOnParts = decPartsValue \* decTAX

' End If

Return decTaxOnParts

End Function

Function TotalCharges(ByVal decP As Decimal, ByVal decSL As Decimal) As Decimal

Dim decTotalCharges As Decimal = 0D

' Add all the charges together

decTotalCharges = decP + decSL + TaxCharges(decP)

Return decTotalCharges

End Function

## Chapter 6 Assignment

### Multilpe Choice (even questions) pages 396-399

1. C
2. B
3. B
4. C
5. C

### True/False (even questions) pages 396-399

1. F
2. T
3. F

### Short Answer (even questions) pages 396-399

1. An *argument* is the specific values passed to a procedure/function. A *parameter* is special variable that receives the value being passed.
2. If you are passing an argument and do not wish for the value to change, pass the argument *By* *Value* instead of *By Reference*. *ByVal* only uses the value store in the memory location and doesn't have access to change what is stored in the location.
3. A function must use a *Return* statement which uses the Return keyword. The Function signature will state the data type to return as well.

### Algorithm Workbench (even questions) pages 396-399

1. intResult = Square(4)
2. Display(intAge, strName, sngIncome)