Please answer questions in the space provided. Question point values are shown in parentheses.

1. (15) Assume you have the following variable declarations and assignments:

```vbnet
Dim A As Integer, B As Integer, C As Integer
Dim D As Integer, E As Integer
Dim X As Double
```

A = 40
B = 10
C = 25
D = 2
E = 100

Given these values, determine the value of X in each of the following assignment statements:

a. \( X = -B^{2}/40 \)

b. \( X = C \text{ Mod } B \times 2 \)

c. \( X = C \text{ Mod } C \text{ \textbackslash } B \)

d. \( X = A + B / D + E \)

e. \( X = -A + -B \times C / D \times E \)
2. (18) For each of the following statements, indicate if it is True or False. Write your answer in the left margin.

   a. All references to a function must be made in the context of an expression, i.e., you cannot reference a function unless it is either all or part of an expression.

   b. When storing something into a String-type variable, it is impossible to generate a type mismatch error.

   c. Although it is not mandatory, it is always a good idea to provide a type (Single, Integer, etc.) when defining a general sub procedure.

   d. A variable declared using the Static statement in a procedure and another variable declared using the Public statement in a Code Module both have the same lifetime.

   e. You can use If…Then…Else statements and the Select…Case statements interchangeably, that is, the only difference is one of style and personal choice.

   f. Using module-level scope for a variable declared with a Dim statement is preferred over local scope even if the variable is only needed in one procedure because then the variable does not have to be recreated each time the procedure is executed.

3. (15) For each of the following, write the correct VB code to accomplish the stated objective.

   a. You have two option buttons named optYes and optNo. Write a code segment (part of a command button’s Click event) that checks the option buttons and sets the integer variable “status” equal to 1 if optYes has been chosen and 2 if optNo has been chosen. You may assume that either one or the other has been chosen.
b. You have two check boxes named chkOvertime and chkHoliday. Write a code segment (part of a command button’s Click event) that tests these check boxes and sets the integer variable “status” equal to one of four values as determined by the following table.

<table>
<thead>
<tr>
<th>chkOvertime</th>
<th>chkHoliday</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unchecked</td>
</tr>
<tr>
<td>unchecked</td>
<td>1</td>
</tr>
<tr>
<td>checked</td>
<td>3</td>
</tr>
</tbody>
</table>

c. You have an option button named optSummer. Write the complete appropriate event procedure that sets the module-level integer variable “season” equal to 4 if the user clicks on the option button to turn it on.
4. (18) Using the code shown below, indicate the output generated by the three Form1.Print statements when the cmdQ4_Click() event is executed.

| Option Explicit | Private Sub doSubA(ByVal x As Integer, _
| Dim a As Integer | y As Integer, ByVal z As Integer) |
| Dim b As Integer | Dim b As Integer |
| ---------------- | b = x |
|                  | y = b |
|                  | z = z + 1 |
| Private Sub cmdQ4_Click() | End Sub |
| Dim a As Integer | Private Sub doSubB(x As Integer, y As Integer) |
| Dim b As Integer | Static c As Integer |
|                  | c = c + 1 |
|                  | b = c |
|                  | x = y + 10 |
|                  | a = 55 |
| doSubA a, b, c | End Sub |
| Form1.Print a, b, c | Private Sub doSubC(a As Integer, _
| doSubB c, b | b As Integer, c As Integer) |
| Form1.Print a, b, c | doSubB a, b |
| doSubC a, b, c | a = b |
| Form1.Print a, b, c | c = c + 1 |
| End Sub | End Sub |
5. (16) The KeyPress event shown below is designed to enforce the following rules:

   a. Make the Enter (Return) key act like the Tab key.
   b. Turn lowercase letters into their uppercase equivalents.
   c. Reject all digits and the Backspace key and beep.
   d. Accept a right parenthesis “)” but only if at least one left parenthesis has already been entered. If rejected then issue a beep.
   e. Accept all other keystrokes.

The code below contains errors (both syntax and logic) that mean the rules expressed above will not be correctly enforced. Find each error and correct it.

Private Sub txtQ5_KeyPress(KeyAscii As Integer)
    Select Case KeyAscii
        Case vbKeyReturn
            KeyAscii = 9
        Case "a" To "z"
            KeyAscii = KeyAscii + 32
        Case 0 To 9, vbKeyBack
            ' do nothing
        Case ")"
            KeyAscii = 0
            Beep
    End Select
End Sub
6. (18) The screen images below show an application that can determine if a number is an even or odd number.

The click event for the “Even or Odd” command button shown above is:

```vbnet
Private Sub cmdEvenOrOdd_Click()
    Dim number As Integer
    number = txtTestValue.Text
    If even(number) Then
        MsgBox number & " is an even number.", vbInformation
    Else
        MsgBox number & " is an odd number.", vbInformation
    End If
End Sub
```

Write the complete sub procedure or function (whichever is appropriate) for the routine “even” referenced in the code above. Remember that all even numbers are evenly divisible by 2.
Answer Key

1. a. -2.5  
   b. 5  
   c. 1  
   d. 145  
   e. -12540

2. a. True  
   b. True  
   c. False – functions, not general sub procedures, have types.  
   d. False – unlike the global variable, the static variable will “die” if the form is unloaded.  
   e. True  
   f. False – you always want to use the narrowest scope possible.

3. a. A sample solution is:

   ```vba
   If optYes.Value = True Then
       status = 1
   Else
       status = 2
   End If
   ```

   b. A sample solution is:

   ```vba
   If chkOvertime.Value = vbUnchecked Then
       If chkHoliday.Value = vbUnchecked Then
           status = 1
       Else ' Holiday is checked
           status = 2
       End If
   Else ' Overtime is checked
       If chkHoliday.Value = vbUnchecked Then
           status = 3
       Else ' Holiday is checked
           status = 4
       End If
   End If
   ```

   c. A sample solution is:

   ```vba
   Private Sub optSummer_Click()
       season = 4
   End Sub
   ```
4. The output from the three `form1.print` statements is

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>12</td>
</tr>
</tbody>
</table>

5. A correct version of the `KeyPress` event is:

```vbnet
Private Sub txtQ5_KeyPress(KeyAscii As Integer)
    Select Case KeyAscii
        Case vbKeyReturn
            SendKeys "{tab}"    ' Send the tab key
            KeyAscii = 0
        Case Asc("a") To Asc("z")
            KeyAscii = KeyAscii - 32
        Case vbKey0 To vbKey9, vbKeyBack
            KeyAscii = 0
            Beep
        Case Asc("(")
            If InStr(1, txtQ5.Text, ")") = 0 Then
                KeyAscii = 0
                Beep
            End If
    End Select
End Sub
```

6. A correct version of the function is:

```vbnet
Public Function even(ByVal x As Integer) As Boolean
    If x Mod 2 = 0 Then
        even = True
    Else
        even = False
    End If
End Function
```