

SLIATE

SRI LANKA INSTITUTE OF ADVANCED TECHNOLOGICAL EDUCATION

(Established in the Ministry of Higher Education, vide in Act No. 29 of 1995)

Higher National Diploma in Information Technology

Second Year, First Semester Examination – 2018

Rapid Application Development

ANSWER

Instructions:

Answer any five (05) questions only.

All questions carry equal marks.

No. of questions : 06

No. of pages : 05

Time : Three (03) hours

Q1.)

- a) Explain the term “Rapid Application Development” and State any two key objective of RAD.

Usable systems are built within a short period of time (as little as 2-3 months)

[04 Marks]

- b) “Traditional software development methodologies have certain issues”.

State and briefly explain three (03) of such issues.

- **Cost and schedule overruns**
- **Product not fit for business**
- **High workload**
- **Projects get cancelled**
- **Friction among managers, developers and customers**

[06 Marks]

- c) Briefly explain two (02) types of classic mistakes.

- i. **People related**
 - ii. **Product related**
 - iii. **Technology related**
 - iv. **Process related**
- should be describe briefly.**

[04 Marks]

- d) Justify the following statements.

- i) “Rapid application development tools supports effective development”

Rapid application development tools support to speed up the development.

- **Cost effective.**
- **Earlier usage**
- **Higher user involvement**

[03 Marks]

ii) "Rapid application development is not suitable for all type of software".

Not suitable to following type projects

Research project, large scale project, and innovative project.

In some project

Higher accuracy expects in some project.

Accuracy estimation required.

Etc..

[03 Marks]

Q2.)

a) Give two operators for following types with examples.

i) Mathematical operators

+ - * /
[Watermark: HNDIT 2-2311]

ii) Relational operators

< > = <= >=
[Watermark: HNDIT 2-2311]

[2x2 Marks]

b) i) Give four (4) logical operators in their precedence order (highest to lowest).

- Not
- And
- Or
- Xor

[4x1 Marks]

ii) $X = 2 * 4 + 6 - 8 / 4 + 2 * (7 \bmod 4)$

What is the value of X after execution of above statement?

- c) Write Visual Basic code to accept an integer values using input box and to display the square value of using message box.

```
Sub Button1_click()
```

```
Dim x,y as integer
```

```
x=inputbox("Please input number")
```

```
y=x*x
```

```
End sub
```

[04 Marks]

- d) Answer the questions given below by considering the following rate of bonus in basic salary and user interface.

| Basic salary range | Percentage of Bonus |
|--------------------|---------------------|
| 10000 and below | 30% |
| 10001-50000 | 20% |
| 50001-100000 | 15% |
| More than 100000 | 10% |

- i. Give the names for each of the above controls according to the Visual Basic naming convention.

Naming convention should be used for every control.
Mainly textboxes(2) and buttons(1) should be named.

Example

Button -btnBonCal / bonCalBtn

Textbox-txtBasicSalary/basicSalaryTxt etc

[02 Marks]

- ii. Write visual basic codes to bonus calculate button click event shown in the interface.

Sub bonCalBtn_click()

```
Dim BSalary As Double
Dim Bonus As Double
BSalary = Val(bSalaryTxt.Text)
If BSalary <= 10000 Then
    Bonus = BSalary * 0.3
ElseIf BSalary <= 50000 Then
    Bonus = BSalary * 0.2
ElseIf BSalary <= 100000 Then
    Bonus = BSalary * 0.15
Else
    Bonus = BSalary * 0.1
End If
```

```
BAmountTxt.Text = Bonus
End sub
```

[04 Marks]

Q3.)

- a) Write the differences of following control structures with examples.

i. If Then Else VS Select Case

Select case more suitable to multi branching

Explain with suitable example

[04 Marks]

- b) Write Visual Basic code to find out a given number is weather prime or not between the range of 1-100.

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim num, n As Integer
```

```

Dim prime As Boolean = True
n = 2
num = Val(txtNum.Text)
If num = 1 Then prime = False
If num = 2 Then prime = True

Do While n <= (num + 1) / 2 And prime
    (Do While n <= 100 And prime)
    If num Mod n = 0 Then
        prime = False
    End If
    n = n + 1
Loop

If prime Then
    MsgBox(" is prime number")
Else
    MsgBox(" is not prime number")
End If

End Sub

```

[06 Marks]

- c) Write the output of following code segments.
- i)

```

For i = 1 To 4
    For j = 1 To i
        TextBox1.Text = TextBox1.Text & i
    Next j
    TextBox1.Text = TextBox1.Text & vbCrLf
Next i

```

[04 Marks]

- ii) Output of button1_click .

```

Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Dim x As Integer = 10
        Dim y As Integer = 15
        Dim z As Integer = 2

        Do While z < 5
            x += 1
            y += 1
            z += 2
        End Do
    End Sub
End Class

```

```

Loop
Select Case x
    Case Is <= 5 :If y Mod 2 <> 0 Then y = y + 5
        x += 5

    Case Is <= 10 :If y Mod 2 <> 0 Then y = y + 10
        x += 10

    Case Is <= 15 :If y Mod 2 <> 0 Then y = y + 15
        x += 15

    Case Else :If y Mod 2 <> 0 Then y = y + 20
        x += 5
End Select

MsgBox(x & " " & y & " " & z)
End Sub
End Class

```

[06 Marks]

Q4.)

a) Describe the followings

i. Procedure (sub)

procedure is a block of Visual Basic statements inside Sub, End Sub statements. Procedures do not return values.

ii. Functions.

Functions are named block programs (procedures) that carry out a specific task and also return a result or value. They are marked by the Function and End Function statements.

[2x2 Marks]

b) Write Visual basic code for a function fact(number), which returns the factorial value of number.(fact(5) returns 1x2x3x4x5=120)

Function fact (by val n as integer)

Dim f as integer=1

For i=1 to n

f=f*i

next i

return f

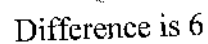
End Function

| | | |
|-----|-------|-----------|
| i.) | ByVal | ii.)ByRef |
|-----|-------|-----------|

ByRef - Pass a reference to original variable

i)

End Class



ii)

```
Private Sub Button1_Click(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles Button1.Click
```

```

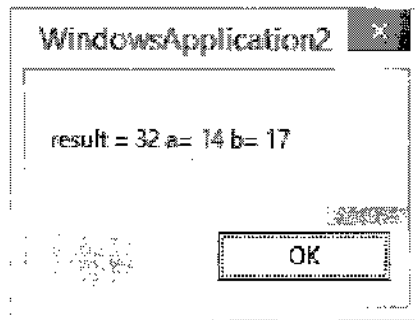
msgBox("result = "& Result(a,b) & "a= " & a & "b= " & b)
End Sub

```

```

function Result(ByVal x as integer, Byref y as integer) As
integer
X=x+1
Y=Y+1
Return x+y
End sub
End Class

```



result=32 1 marks
a=14 1 marks
b=17 2 marks

[4 marks]

Q5.)

- a) Briefly explain the terms 'class' and 'object'.

A class is a blueprint that describes an object and defines attributes and operations for the object.

An object is an instance of a class.

[04 Marks]

- b) Write a VB code for following scenarios.

- i. Create Dog class with attributes of name-String. Which is not accessible outside the class.

Class Dog

Private name as string

End class

[2 marks]

- ii. Dog class has Methods

- setName(name)- to assign a name to dog.

- showName()-to display the name by message box.

```
Public sub setName(byval na as string)
```

```
    Name=na;
```

```
End sub
```

```
Public sub showName()
```

```
    MsgBox(name)
```

```
End sub
```

[4 marks]

c) Explain following terms with suitable a class example

i) Constructor of a class

Constructor is method. This method is executes during object initializing time.

```
Sub New()
```

```
    Name="no name"
```

```
End sub
```

ii) Method/function overloading

- You can define method or property multiple times with different argument list

```
Public sub setName(byval na as string)
```

```
    Name=na;
```

```
End sub
```

```
Public sub setName()
```

```
    Name="I am";
```

End sub

[4 Marks]

d) What is the output of the following code segment?

[06 Marks]

```
Public Class Car
    Public Overridable Sub Accelerate ( )
        System.console.writeline ("Speed increase by 10 km/h")
    End Sub
    Public Sub BreakFunction( )
        System.console.writeline ("Speed reduced to 0 km/h")
    End Sub
End Class

Public Class RaceCar
    Inherits Car
    Public Overrides Sub Accelerate ( )
        System.console.writeline("Speed increase by 20 km/h ")
    End Sub
End Class

Module Module1
    Sub main()
        Dim sc As RaceCar= New RaceCar ( )
        Dim c as Car=new Car()
        sc. Accelerate()
        c. Accelcrate()
        sc. BreakFunction ()
    End sub
End Module
```

output

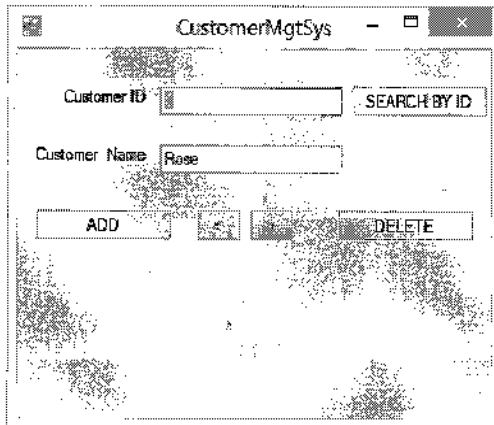
Speed increase by 20 km/h

Speed increase by 10 km/h

Speed reduced to 0 km/h

Q6.)

The following prototype interface was created using VB.Net to facilitate customer name keeping purpose. The back end for the system has been created using MS Access.



- a) "ADO.NET is Disconnected Data Access Architecture"

Explain the above statement?

By keeping connections open for only a minimum period of time, ADO .NET conserves system resources and provides maximum security for databases and also has less impact on system performance

[2 Marks]

- b) Following code segment for connecting above interface with the MS Access and load first record of table to fill data into textboxes. Fill in the blanks with suitable term.

```
Public Class CustomerFrm
    Private oledbCon As OleDb.OleDbConnection
    Private oledbDAdb As OleDb.OleDbDataAdapter
    Private ds As DataSet
    Private oleConString As String
    Private rec As Integer = 0

    Private Sub CustomerFrm_Load(ByVal sender As System.Object, ByVal e As
    System.EventArgs) Handles MyBase.Load
        oleConString = "Provider=Microsoft.Jet.OLEDB.4.0;
                        DataSource=C:\Users\Acer\Documents\customerData.mdb"
        'To initialize of connetion String
```

```

Dim sql As String = i"SELECT * FROM customer;"
'sqlstatement to get whole data of table customer

oledbCon = New OleDb.OleDbConnection(ii. oleConnectionString)
'To initialize oledb connection

oledbDAdb = New OleDb.OleDbDataAdapter(iii sql, oledbCon)
'To initialize oledb data adapter

ds = New DataSet
'To initialize data set
Try

oledbCon.Open
'To open a oledb connection

v_oledbDAdb.Fill(ds, "cust")
'To fill the data into dataset

oledbCon.Close()
'To close a oledb connection

Catch ex as exception
MsgBox("Problem in connection" & ex.message)

End Try

idTxt.Text = ds.Tables("cust").vi rows(0).item(0)_____
nameTxt.Text = ds.Tables("cust").vii rows(0).item(1)_____

End Sub
End Class

```

[1x7 Marks]

viii. Briefly explain purpose of try catch using in above code.

Provides a way to handle some or all possible errors that may occur in a given block of code, while still running code.

[2 Marks]

- c) Write the code segment to save the information displayed on the form into the database.(Add Button)

Assume that the database contains a table with the following structure:

Customer (cId-integer, cName-text)

```

Private Sub addbtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles addbtn.Click
    Dim insertSql As String = "INSERT INTO Customer VALUES(" & Val(idTxt.Text) & "," &
    nametxt.Text & "');"

    Dim OLEDBCOM As New OleDb.OleDbCommand(insertSql, oledbCon)
    Try
        oledbCon.Open()
        OLEDBCOM.ExecuteNonQuery()
        oledbCon.Close()
    Catch EXC As Exception
        MsgBox(EXC.Message)
    End Try
End Sub

```

| Customer | |
|----------|--------|
| ID | Name |
| 0 | Rose |
| 1 | Mery |
| 2 | Perera |
| 3 | Mohan |
| 5 | Ramesh |

[05 Marks]

- d) Write the code segment to get the next record information of a customer. (> Button).

```

Private Sub nextBtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles nextBtn.Click
    If rec < ds.Tables("cust").Rows().Count - 1 Then rec = rec + 1
    idTxt.Text = ds.Tables("cust").Rows(rec).Item(0).ToString
    nametxt.Text = ds.Tables("cust").Rows(rec).Item(1).ToString
End Sub

```

[04 Marks]