higher education & training

Department:
Higher Education and Training
REPUBLIC OF SOUTH AFRICA

MARKING GUIDELINE

NATIONAL CERTIFICATE (VOCATIONAL)

INTRODUCTION TO SYSTEMS DEVELOPMENT
(Second paper)
NQF LEVEL 2

25 November 2022

This marking guideline consists of 5 pages.

Copyright reserved

QUESTION 1



1-14 **[30]**

QUESTION 2

2.1 Examination number (1)

2.2 2.2.1

```
strSurname = txtSurname.Text
appointmentDate = CDate (dtpDate.Text)
appointmentTime = CDate(dtpTime.Text)
```

(4)

2.2.2

```
Select Case cmbSpecialist.SelectedIndex
    Case 0
        intConsultationFee = 200
    Case 1
        intConsultationFee = 500
    Case 2
        intConsultationFee = 600
    Case Else
        intConsultationFee = 700
End Select
```

(6)

2.2.3
```
decAmountPayable✓ = intAppointmentFee +
intConsultationFee✓
```
(2)

2.2.4
```
lblConsultationFee.Text✓=FormatCurrency✓
(CStr(intConsultationFee)) ✓
```
(3)

2.2.5
```
lblAmountPayable.Text✓ = FormatCurrency✓
(CStr(decAmountPayable))✓
```
(3)

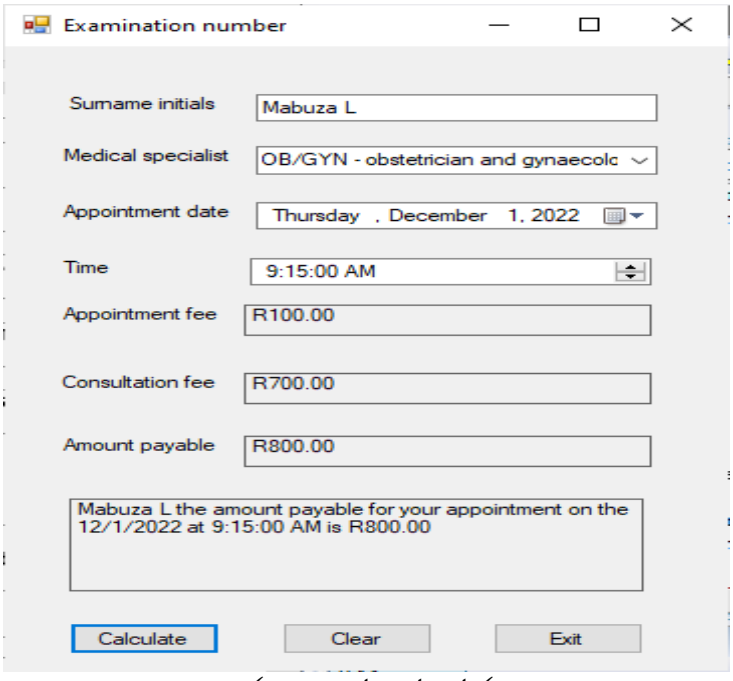
2.2.6
```
lblDisplay.Text = strSurname✓ & " the amount payable for
your appointment on the " & appointmentDate✓ & " at " &
appointmentTime✓ & " is " & FormatCurrency
(decAmountPayable)✓
```
(4)

2.3
```
txtSurname.Clear()✓
cmbSpecialist.Text = ""✓
lblAppointmentFee.Text = ""
lblConsultationFee.Text = ""✓
lblAmountPayable.Text = ""
lblDisplay.Text = ""
txtSurname.Focus()
```
(3)

2.4
```
If MessageBox.Show✓ ("Do you want to exit the program", "Exit",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) ✓ =
Windows.Forms.DialogResult.Yes✓ Then
        Me.Close()✓
    End If
```
(4)

2.5 Form title bar EXAMINATION NUMBER. (1)

2.6



✓ correct output✓ (2)

2.7 Print source code with EXAMINATION NUMBER.✓ (1)

[34]

QUESTION 3

3.1 Examination number (1)

3.2 3.2.1

```
If Not IsNumeric(txtFirst.Text) Or Not
        IsNumeric(txtSecond.Text) Then✓
         If Not IsNumeric(txtFirst.Text) Then✓
           MessageBox.Show("The first number must be a
           numeric value", "Error", MessageBoxButtons.OK,
           MessageBoxIcon.Error)
                txtFirst.Clear()✓
                txtFirst.Focus()
            Exit Sub✓
```

(4)

3.2.2

```
elseIf Not IsNumeric(txtSecond.Text) Then✓
        MessageBox.Show("The second number must be a
       numeric value", "Error", MessageBoxButtons.OK,
       MessageBoxIcon.Error)
          txtSecond.Clear()✓
          txtSecond.Focus()
       Exit Sub✓
    End If
End if
```

(3)

3.2.3

```
intFirstNumber✓ = CInt(txtFirst.Text)✓
intSecondNumber = CInt(txtSecond.Text)✓
```

(3)

3.2.4

```
If intFirstNumber > intSecondNumber Then✓
       intHighestNumber = intFirstNumber✓
Else✓
       intHighestNumber = intSecondNumb✓
End If
```

(4)

3.2.5

```
decDivision✓ = CDec(Math.Truncate✓ (intFirstNumber/
intSecondNumber))✓
```

(3)

3.2.6

```
intRemainder = intFirstNumber Mod✓ intSecondNumber✓
decAverage = CInt(Math.Round✓ (intFirstNumber +
intSecondNumber) ✓ / 2) ✓
```

(5)

3.2.7

```
lstDisplay.Items.Add("The highest number is: " &
intHighestNumber)✓
lstDisplay.Items.Add("Division is: " & decDivision)✓
lstDisplay.Items.Add("The remainder is: " & intRemainder)✓
lstDisplay.Items.Add("")✓
lstDisplay.Items.Add("Average is: " & decAverage)✓
```

(5)

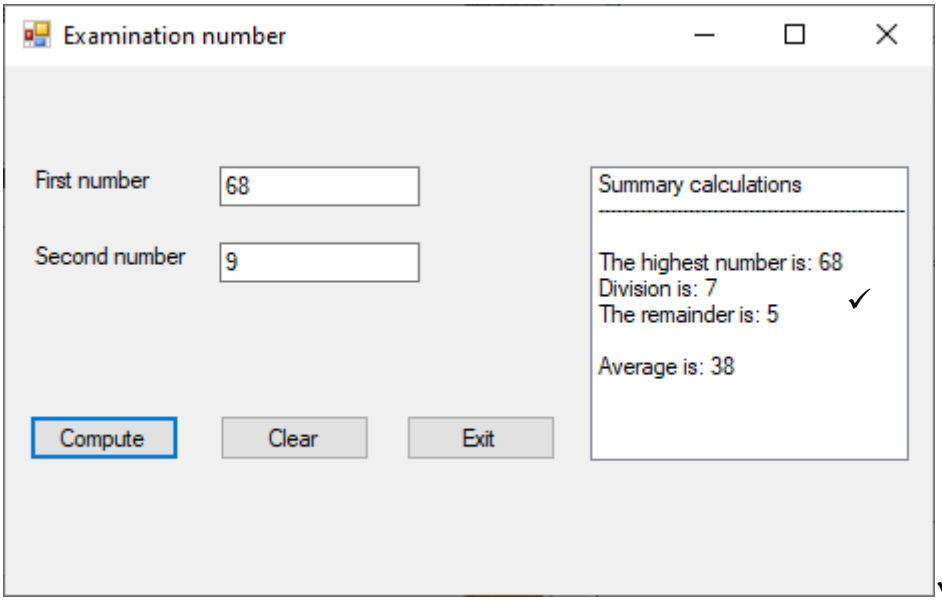
Copyright reserved

3.3
```
txtFirst.Clear()✓
txtSecond.Clear()✓
txtFirst.focus()✓
```
(3)

3.4 `Me.close()`✓ (1)

3.5 Form title bar✓ (1)

3.6



✓ (2)

3.7 Print code. (1)

[36]

TOTAL: 100

Copyright reserved