

ICT3611

May/June 2014
Mei/Junie 2014

ADVANCED GRAPHICAL USER INTERFACE PROGRAMMING
GEVORDERDE GRAFIESE GEBRUIKERSKOPPELVLAKPROGAMMERING

Duration : 2 Hours
Tydsduur : 2 Uur

75 Marks
75 Punte

EXAMINATION PANEL AS APPOINTED BY THE DEPARTMENT.
EKSAMENPANEEL SOOS DEUR DIE DEPARTEMENT AANGEWYS.

Closed book examination.
Toeboekeksamen.

This examination question paper remains the property of the University of South Africa and may not be removed from the examination venue.
Hierdie eksamenvraestel bly die eiendom van die Universiteit van Suid-Afrika en mag nie uit die eksamenlokaal verwyder word nie.

Examiners	:	
First	:	Dr W van Staden
Second	:	Mr D Bischoff
External	:	Mr AAK Buitendag (TUT)

ANSWERS/SOLUTIONS

Level 1 = {F - Functional, C - Comprehension}

Level 2 = {I - Investigative, A - Analysis}

Level 3 = {G - Generative, S - Synthesis}

Instructions:

1. Answer all questions.
2. Write neatly and legibly.
3. This paper consists of 7 pages.

Question 1 Error Handling [8]

- (a) Explain what is meant by “structured error handling” in VB.NET. [2]

Solution: (C)

Structured error handling in VB.NET makes use of try..catch blocks to handle errors

- (b) You are provided with a function called `summation(n As Integer) As Integer` in a library. This function calculates the sum of the first n negative integers. Since, the creator of the function ensured that it cannot be called with a positive integer it throws a `PositiveIntegerException` when it is called with a positive valued parameter. Write a snippet of code which will call the `summation` function, making sure to show a message in the event of an error. [6]

Solution:

(G)

```

1   Dim m As Integer
2   Dim n As Integer
3
4   Try
5       m = factorial(n)
6   Catch e As PositiveIntegerException
7       MsgBox("Cannot pass a non-negative integer...")
8   End Try

```

Question 2 Modules and Procedures [5]

- (a) Differentiate between declaring a form variable **public** and declaring it **private**. [2]

Solution: (F)

Public variables can be accessed by any calling statement, and private variables can only be called by statements from within the same block of code (the form)

- (b) Clearly explain why one would place a function procedure in a module as opposed to placing it in a form. Does this mean we should never place function procedures

Question 3 Files and String manipulation [28]

- (a) Declare a record in VB syntax which will be used to store a student's full name (no longer than 100 characters), a module (course) code (no longer than 7 char-

in forms? Clearly explain your answer. [3]

Solution:

(A)

Functions are placed in modules if they should have global scope. No, functions are placed in forms if the function relates to the functionality of that form only.

acters), and their year and examination mark.

[6]

Solution: (A)

```

1  Structure  ModuleMark
2      <VBFixedString (100)> Public name as String
3      <VBFixedString (7)> Public module as String
4      Public year_mark As Double
5      Public exam_mark As Double
6  End Structure

```

(b) Consider the following content of a text file:

```

1  Saunders, H, 23
2  Vahed, P, 45
3  Naidoo, A, 23
4  Mabila, Q, 20

```

Values are separated with a comma, and the fields are: surname, initial, age.

- i. Create a function procedure `CountAges` which accepts an integer, and string as parameters and returns an integer. The integer parameter represents an age, and the string parameter is the name of a file on disk. The function scans the file (which has the same format as provided above) and counts all the entries that have an age matching the integer parameter passed to it. For example, the function returns 2 if called with 23 (using the data provided above). You may assume the file exists (there is no need to determine if it

exists).

[13]

Solution:

(S)

Any one of the following up to a maximum of 13 points.

```

1
2  Public Function  countAges(_
3      ByVal age as Integer , ByVal filename As String ) _
4      As Integer
5
6      Dim count As Integer = 0
7      Dim strmrdr As System.IO.StreamReader
8
9      'Open the file for reading
10     strmrdr = new System.IO.StreamReader(fileName)
11
12     'Read one line of text until the end of the file
13     Do Until strmrdr.EndOfStream
14         Dim line As String = strmrdr.ReadLine()

```

```
15
16     'Split the line of text into words
17     Dim words() As String = line.Split(',')
18
19     If Cint(words(2)) = age Then
20         count += 1
21     Endif
22 Loop
23
24     'close the file
25     strmrdr.close()
26
27     return count
28 End Function
```

- ii. Write a function procedure called `AgeFrequency` which accepts a file name,

and an integer as parameters and returns the age frequency for the integer parameter. The age frequency is the number of times an age occurs in a file divided by the number of entries in the file (the format of the file will be as provided at the start of this question). Assume a function called `CountAges` exists. `CountAges` accepts an age (integer) and a file name (string) as parameters and returns an integer indicating the number of times the age appears in the file. Also assume a function called `CountEntries` exists which accepts a file name (string) as parameter and returns the number of entries (lines) in the file (the format of the file will be as provided at the start of this question). Use the `CountAges`, and `CountEntries` functions described to write the `AgeFrequency` function. (Hint: Start by writing down the steps you need to

```

17     'If we don't worry about divide by zero...
18     return age_counter / entry_counter
19
20 End Function

```

- iii. There is an obvious error that can occur when performing the frequency calculation, what is this error? How would you avoid it? [2]

Solution:

(S)

There is a divide by zero error that can happen . We can use a try catch, or just return 0.

take to get a working solution). Do not assume the file exists. [7]

Solution:

(G,S)

```

1
2 Public Function AgeFrequency(_
3     ByVal age As Integer,
4     ByVal FileName As String) As Double
5
6     Dim entry_counter As Integer = 0
7     Dim age_counter As Integer = 0
8
9     If (System.File.Exists(FileName )) Then
10        entry_counter As Integer = CountEntires(Filename)
11        age_counter As Integer = _
12            CountAges(age, Filename)
13    Else
14        MsgBox("File does not exist")
15    Endif
16

```

Question 4 Classes and Objects [25]

- (a) Explain why it is generally not a good idea to store user input in GUI control value fields, and how this problem can be avoided (aside from disabling editing on the control). [2]

Solution:

(C)

The values may change while the application is busy performing calculations with them. Create backing classes/variables that store the values is much safer.

- (b) Construct a simple design for an application that manages the renting of boats to customers at a marina. Make use of the Model-View-Controller paradigm in your design. (Hint: Make sure to allow boats to be assigned to customers. Provide a diagram to illustrate your design.). Explain the function of each element of your design. [6]

Solution: (G)

Any design which shows a boat and customer class, including a controller. Customers and Boats are Plain Old Data (POD) objects, and the Controller is used to manage assignment of attendees to concerts.

- (c) Write a snippet of code which defines an abstract base class to represent Cars at a car rental agency. Provide a property for the car's type (sedan, hatch-back, SUV, 4x4, and so on). It should not be possible to change the car's type. For rental cars the base fare is calculated differently for four-wheel drive and sedans: provide an abstract function definition called `BaseFare` which will calculate the base fare (it takes an integer representing the number of days the car will be rented). [14]

Solution: (G)

```

1  Class MustInherit Car
2      Protected Dim _type As String
3
4      Public Readonly Property CarType() As String
5          Get
6              Return _type
7          End Get
8      End Property
9
10     Public MustOverride
11         Function BaseFare (days As Double) As Double
12
13 End Class

```

- (d) Clearly explain what a default constructor is in VB, and provide a snippet of code to show what one looks like. [3]

Solution:

(S)

A default constructor is a constructor that accepts no parameters

```
1 Public Sub New( )  
2 End Sub
```

Question 5 ADO.NET [9]

- (a) Explain the mechanism ADO.NET uses to ensure that data can be accessed in a uniform way. [3]

Solution:

(C)

ADO.NET uses XML to store and transport data, in this way, data from multiple sources can be accessed without care for their format.

- (b) Explain how a DataGridView control works on a high-level. [3]

Solution:

(A)

Presents information using a grid of rows and columns. It does this by using a data adapter and dataset in order to access data.

- (c) Explain why one has to issue a command in ADO.NET to write changes in the dataset to the database. What is this type of arrangement called? [3]

Solution:

(A)

Because the dataset only represents the data in the database – to change the actual database, you need to issue a command to write the changes to the database. This is called a disconnected database.

Total: 75