The Open University

MT264/Specimen

Course Examination
Designing applications with Visual Basic

Time allowed: 3 hours

There are **TWO** parts to this examination. You should attempt **BOTH**.

You should attempt **ALL** the questions in Part 1.

In Part 2 there are three questions; marks will be given for your best **two** answers.

You are advised to spend about 100 minutes on Part 1 and 70 minutes on Part 2, and to leave yourself about 10 minutes for checking. Part 1 carries 64% of the total marks and Part 2 carries 36%.

You are advised to show all your working and to give reasons for all your answers unless a question is explicitly phrased 'Write down . . . '. You should begin each answer on a new page of the answer book.

At the end of the examination

Check that you have written your personal identifier and examination number on **each** answer book used. **Failure to do so will mean that your paper cannot be identified.**

Put all your used answer books together with your signed desk record on top. Fasten them in the top left corner with the round paper fastener. Attach this question paper to the back of the answer books with the flat paper clip.

The use of calculators is not permitted in this examination.

Copyright © 2009 The Open University **WEB 98569 6**

1.1

Part 1

*You should attempt **ALL** the questions in Part 1 and are advised to spend about **100 minutes** on it. This part carries 64% of the marks for the whole examination.*

There are eight questions, each worth 8 marks, in Part 1.

Question 1 (8 marks)

(a) Write down design code to make the Text property of a text box aTextBox have the value "Hello world".

(b) What is the effect of the following design code?

```
Set aButton.BackColor To Color.Red
```

(c) Write down design code for the following tasks:

- declare an integer variable maxSize;
- initialise maxSize to the value 100;
- set the Maximum property of a NumericUpDown control called aNumUpDn to maxSize.

(d) Which property of a timer determines the frequency of its tick events?

Question 2 (8 marks)

Suppose that there is a class Book with the following description.

Class table: Book

Properties

ISBN **As** String

Title **As** String

Author **As** String

NumberSold **As** Integer

Constructor

New

Method

salesOver(salesTarget **As** Integer) **As** Boolean

```
Constructor New
    'An object of type Book is created, with ISBN, Title and Author set to the
    'empty string and NumberSold set to 0.
End Constructor

Method salesOver(salesTarget As Integer) As Boolean
    'Preconditions: salesTarget is not negative.
    'Postconditions: If NumberSold is more than salesTarget then True is
    'returned. Otherwise False is returned.
End Method
```

(a) What is the method identifier of the only method given above?

(b) Does this method have any parameters? If so, what type are they?

(c) Does this method return a value? If so, what is the type of the return value?

(d) Can this method change the state of an object of type Book on which it is invoked?

(e) (i) Write design code to declare and create an object aBook of type Book with Title property set to "A new book" and Author property set to "A.N.On".

(ii) What value will be stored by the variable goodSales after execution of the following design code fragment?

```
goodSales As Boolean
'The design code you have written for (i).
Set goodSales To aBook.salesOver(1000)
```

Question 3 (8 marks)

Suppose that the following design code has been executed.

```
aDictionary As Dictionary(Of String, Boolean)
key As String
Set aDictionary To New Dictionary(Of String, Boolean)
For index As Integer From 0 To 10
    Set key To "k" + index.toString()
    If index < 5 Then
        aDictionary.add(key, True)
    Else
        aDictionary.add(key, False)
    End If
End For
```

(a) Write down the value of aDictionary.Count.

(b) Write down the return value of aDictionary.containsKey("k12").

(c) Write down the value of aDictionary.Item("k2").

(d) Write down the list of items contained in aList after the following design code is executed.

```
aList As List(Of String)
Set aList To New List(Of String)
For Each keyItem As String In aDictionary.Keys
    If aDictionary.Item(keyItem) = True Then
        aList.add(keyItem)
    End If
End For
```

Question 4 (8 marks)

This question concerns the following class.

Class table: TextSearcher

Property

Text **As** String

Constructor

New

Method

howMany(word **As** String) **As** Integer

```
Constructor New
    'A TextSearcher object is created, with Text set to "".
End Constructor

Method howMany(word As String) As Integer
    'Preconditions: word is not the empty string.
    'Postconditions: The number of occurrences of word in Text is returned.
End Method
```

(a) Write design code for the method howMany. There is no need to include a comment.

You may wish to use the String method

substring(position **As** Integer, len **As** Integer)

This method returns the string that is formed from the string on which it has been invoked by taking the substring starting at index position and which has length len.

(b) Assume that Text is "Not to be abed after midnight is to be up betimes." Choose three different examples of word with which to test your code. Briefly explain what aspect of your code each example is intended to test.

Question 5 (8 marks)

This question concerns the following enumerated type SoilType, and two classes Field and PeaField.

```
Enum SoilType
    Acidic
    Alkaline
    Neutral
End Enum
```

Class table: Field

Properties

Area **As** Double

Shape **As** String

Constructor

New

Method

timeNeededToHarvest() **As** Double

```
Constructor New
    'A Field object is created, with Area set to 0.0 and Shape set to "".
End Constructor

Protected Method timeNeededToHarvest() As Double
    'Preconditions: none
    'Postconditions: A value is returned giving an estimate of the time (in hours)
    'needed to harvest a crop from the field based on the area and shape.
End Method
```

Class table: PeaField Inherits Field

Property
Soil **As** SoilType
Constructor
New
Methods
timeNeededToHarvest() **As** Double Overrides
yieldPerHour() **As** Double

```
Constructor New
    'A PeaField object is created, with Area set to 0.0, Shape set to ""
    'and Soil set to Neutral soil type.
End Constructor

Overrides Method timeNeededToHarvest() As Double
    'Preconditions: none
    'Postconditions: A value is returned giving an estimate of the time (in hours)
    'needed to harvest a pea crop from the field based on the area and shape.
End Method

Method yieldPerHour() As Double
    'Preconditions: none
    'Postconditions: A value is returned giving an estimate of the yield
    '(in kilograms) for each worker–hour spent harvesting.
End Method
```

(a) What is the effect of declaring as **Protected** the method timeNeededToHarvest of the class Field?

(b) (i) Can the method timeNeededToHarvest be invoked by client code on an instance of PeaField? Explain your answer.

(ii) Can the method yieldPerHour be invoked by client code on an instance of PeaField? Explain your answer.

(c) Write design code to implement the constructor of PeaField. There is no need to include a comment.

Question 6 (8 marks)

(a) Write design code for the body of an event handler that enables users to select a file for opening. You may assume that an OpenFileDialog control openDialog has been added to the form. In addition, your code should:

- ► set the Filter property of openDialog so that only an image file of type .jpg can be chosen;
- ► ensure that if a valid file is selected, then the Image property of a picture box aPictureBox is set to the image stored by that file.

There is no need to include a comment for the event handler.

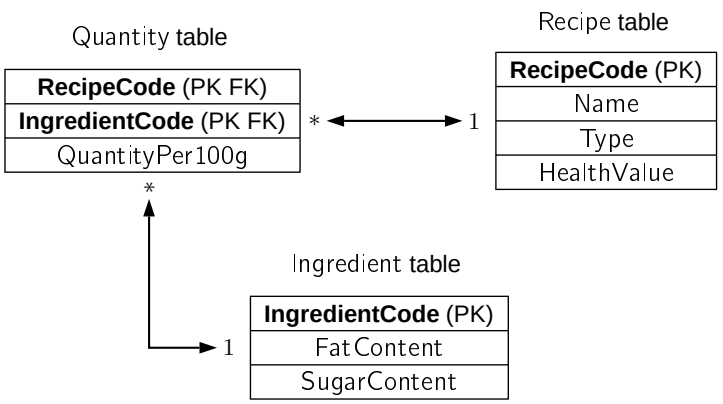
(b) Write design code for the body of the OnPaint event handler of a panel called canvasPanel. The code should:

- ► draw a black (unfilled) square of size (100,100) in the centre of the panel;
- ► write, in blue, the word 'Hello' with the top left of the string at the top left point of the square.

You may assume that the panel is big enough and that there is a font fCurrentFont available. You may use stock pens and brushes. There is no need to include a comment for the event handler.

Question 7 (8 marks)

This question is about a database with the following ER diagram.



Some sample data is given in the tables below.

Recipe

RecipeCode	Name	Type	HealthValue
rec1	Chestnut casserole	vegetarian	green
rec2	Blackcurrant tart	dessert	amber

Quantity

RecipeCode	IngredientCode	QuantityPer100g
rec1	yogurt	20
rec1	apple	30
rec2	butter	10

Ingredient

IngredientCode	FatContent	SugarContent
yogurt	20	0
butter	50	0
apple	0.5	20

(a) Write an SQL statement that returns a table giving the recipe code and name of all the recipes whose Type is vegetarian.

(b) Describe the result table that will be returned by the following SQL statement.

```
SELECT Quantity.RecipeCode
FROM Ingredient INNER JOIN Quantity
ON Ingredient.IngredientCode = Quantity.IngredientCode
WHERE Quantity.QuantityPer100g > 0 AND Ingredient.SugarContent > 0;
```

Question 8 (8 marks)

You have been asked to modify an application for a user unable to use a mouse.

(a) Why might you hope that the application had been designed using an MVC pattern?

(b) The application's current graphical user interface has two labels, a text box for text entry, a button to confirm the text entry and a read-only text box. What steps would you take to ensure that the application could be used without a mouse? Give details of the properties that you would set at design time.

Part 2

*In Part 2, marks will be given for your best **two** answers, and you are advised to spend about **70 minutes** on it.*

Each question carries 18% of the marks for the whole examination, and an indication of the allocation of marks within each question is given in the margin.

Question 9 (18 marks)

This question is about an application to support stock control. The descriptions of the two classes to be used for the model are given below.

Class table: Stock

Properties

Name **As** String

Quantity **As** Integer

Constructor

New

```
Constructor New
    'A Stock object is created, with Quantity set to 0 and Name set to "".
End Constructor
```

Class table: StockAdmin

Field

fStockItems **As** Dictionary(**Of** String, Stock)

Property

StockList **As** List(**Of** String) ReadOnly; no field

Constructor

New

Methods

isCurrentCode(code **As** String) **As** Boolean

numberInStock(code **As** String) **As** Integer

newStockItem(code **As** String, aName **As** String)

addStock(code **As** String, aNumber **As** Integer)

removeStock(code **As** String, aNumber **As** Integer)

'Other methods that are of no concern in this question.

```
Constructor New
    'A StockAdmin object is created, with an empty dictionary of stock items.
End Constructor

ReadOnly Property StockList As List(Of String)
    'Gives a list of the keys in the dictionary of stock items.
End Property
```

```
Method isCurrentCode(code As String) As Boolean
    'Preconditions: none
    'Postconditions: If code is a key in the dictionary then True is returned.
    'Otherwise False is returned.
End Method

Method numberInStock(code As String) As Integer
    'Preconditions: code is a key of the dictionary.
    'Postconditions: The quantity of the stock item whose key value is code
    'is returned.
End Method

Method newStockItem(code As String, aName As String)
    'Preconditions: code is not the empty string and is not already used for
    'a stock item.
    'Postconditions: A new stock item with name aName and quantity set
    'to 0 is added to the dictionary with key set to code.
End Method

Method addStock(code As String, aNumber As Integer)
    'Preconditions: code is a key of the dictionary and aNumber is greater
    'than or equal to 0.
    'Postconditions: The quantity of the stock item whose key value is code
    'is increased by aNumber.
End Method

Method removeStock(code As String, aNumber As Integer)
    'Preconditions: code is a key of the dictionary, aNumber is greater than or
    'equal to 0, and numberInStock(code) is greater than or equal to aNumber.
    'Postconditions: The quantity of the stock item whose key value is code
    'is decreased by aNumber.
End Method
```

(a) Write design code for the methods isCurrentCode, numberInStock, addStock and newStockItem. There is no need to include comments. [*6*]

(b) Suppose that a form in the graphical user interface of the application has two text boxes, codeTextBox and nameTextBox, into which the code and name for a new stock item can be entered, respectively. It also has a button newStockButton for confirming these entries for a new stock item. An instance fStockAdmin of StockAdmin has been declared as a private field of the form.

(i) Briefly describe what the OnClick event handler of newStockButton should do.

(ii) Write design code for the body of the OnClick event handler. There is no need to include a comment. [*6*]

(c) Suppose that the application needs to allow a user to reduce the quantity of a stock item. To implement this, the code will need to invoke the method removeStock.

(i) List the types and names of controls (excluding labels) that you would use for this part of the graphical user interface; your design should ensure that the preconditions of the method removeStock are satisfied. Briefly explain how you would set up your controls so that the preconditions are satisfied.

(ii) Write design code for the event handler that deals with reducing the quantity of a stock item. There is no need to include a comment. [*6*]

Question 10 (18 marks)

This question is about the model for a simple 'jack-in-the-box' game for a small child. The project specification is as follows.

Project specification: Jack-in-the-box

This application should provide two buttons, *Jump* and *Hide*. When the *Jump* button is pressed, a sound file *squeal.wav* should be played and a figure on a spring should move vertically up the screen from behind a square shape. The figure on the spring should stop moving when all of the spring is just above the square shape. When the *Hide* button is pressed, the figure on the spring should move vertically down the screen and hide behind the square shape.

The figure on the spring will be implemented using a sprite of type JumpingJack that inherits HVSprite. The class description is given below.

Class table: JumpingJack Inherits HVSprite

Properties	
NoiseFile **As** String	ReadOnly
Picture **As** Bitmap	ReadOnly
GoingUp **As** Boolean	This is True if the next move of the sprite is upwards and False otherwise.
Constructors	
New	
New(noise **As** String, pos **As** Point, pic **As** Bitmap)	
Methods	
draw(g **As** Graphics)	Overrides
dispose()	

Constructor New
'An instance of JumpingJack is created, with NoiseFile set to the empty
'string and Picture set to a default bitmap. GoingUp is set to True.
'Position is set to (0,0).
End Constructor

Constructor New(noise **As** String, pos **As** Point, pic **As** Bitmap)
'An instance of JumpingJack is created, with NoiseFile set to noise, Position
'set to pos and Picture set to pic. Note that Picture gives both the figure
'and the spring. GoingUp is set to True.
End Constructor

Overrides Method draw(g **As** Graphics)
'Preconditions: none
'Postconditions: Picture is drawn at Position using g.
End Method

Method dispose()
'Preconditions: none
'Postconditions: The bitmap is disposed of.
End Method

There will be a class JackAdmin with the following description.

Class table: JackAdmin

Fields	
fJack **As** JumpingJack	
fBox **As** Rectangle	
Properties	
GameArea **As** Bitmap	ReadOnly
Moving **As** Boolean	ReadOnly
Constructors	
New	
New(area **As** Size)	
Methods	
startJump()	
startHide()	
move()	
updateImage()	
dispose()	

Constructor New
'An instance of JackAdmin is created, with the jumping jack sprite initialised
'using existing noise and picture files (the details of the initialisation are of no
'concern in this question). GameArea is set to a default bitmap, and the box
'is created to fit on the bitmap. Moving is set to False. GameArea is updated.
End Constructor

Constructor New(area **As** Size)
'An instance of JackAdmin is created, with the jumping jack sprite initialised
'using existing noise and picture files (the details of the initialisation are of no
'concern in this question). GameArea is set to a bitmap of size area, and the
'box is created to fit on the bitmap. Moving is set to False. GameArea is
'updated.
End Constructor

Method startJump()
'Preconditions: none
'Postconditions: Moving is set to True and the jumping jack sprite is set to
'be moving upwards, 5 pixels at a time. The jumping jack sprite's sound file
'is played. GameArea is updated.
End Method

Method startHide()
'Preconditions: none
'Postconditions: Moving is set to True and the jumping jack sprite is set to
'be moving downwards, 5 pixels at a time. GameArea is updated.
End Method

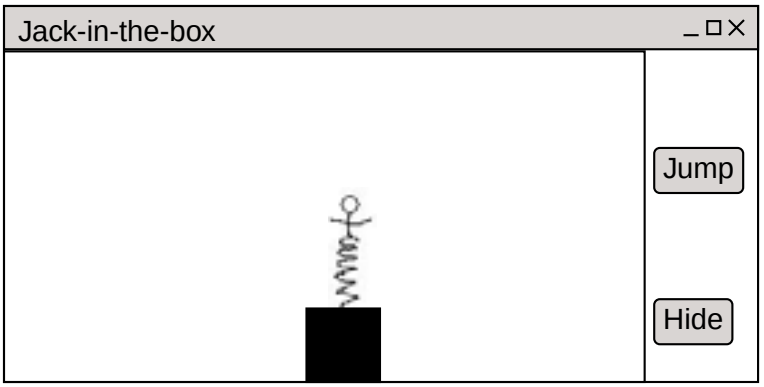
Method move()
'Preconditions: none
'Postconditions: If Moving is True, the jumping jack sprite is moved.
'Checks are made on its position, and Moving is set to False if appropriate.
'Otherwise nothing is done. GameArea is updated.
End Method

```
Private Method updateImage()
    'Preconditions: none
    'Postconditions: GameArea is updated to match the state of the jumping
    'jack sprite.
End Method

Method dispose()
    'Preconditions: none
    'Postconditions: GameArea and the jumping jack sprite are disposed of.
End Method
```

(a) Write design code for the methods updateImage, startJump and move of JackAdmin. There is no need to include comments. [*12*]

(b) A possible graphical user interface and property table for the application are given below.



Property table: Jack-in-the-box

	Name	Property	Initial value
Container			
Form	MainForm	Text	Jack-in-the-box
Controls			
PictureBox	gamePictureBox		
Button	jumpButton	Text	Jump
Button	hideButton	Text	Hide
Component			
Timer	gameTimer	Interval	100
		Enabled	False
Field			
JackAdmin	fJackAdmin		

It has been decided that the timer should only be enabled while the 'Jack' is moving.

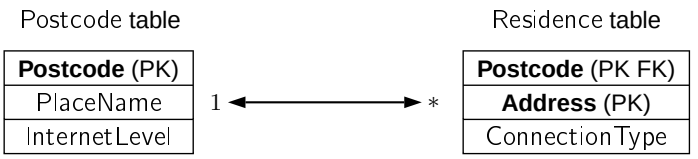
(i) Write design code for a form method updateView that ensures that the picture box display represents the current state of the model.

(ii) Write the design code body for the OnClick event handler of the *Jump* button. There is no need to include a comment.

(iii) Write the design code body for the OnTick event handler of gameTimer. There is no need to include a comment. [*6*]

Question 11 (18 marks)

This question is about an application that gives access to a database PostcodeData containing postcodes. The data records the level of internet connection available at the postcodes, and then for each residence with a postcode, it records the type of internet connection, if any, that is used. The ER diagram is shown below.



Some sample data is as follows.

Postcode

Postcode	PlaceName	InternetLevel
Y95 4QA	York	Broadband available
DN43 2DE	Doncaster	Dial-up

Residence

Postcode	Address	ConnectionType
Y95 4QA	27 Broad Avenue	Broadband
Y95 4QA	25 Broad Avenue	No internet connection
DN43 2DE	41 Narrow Street	Dial-up

The specification of the application is as follows.

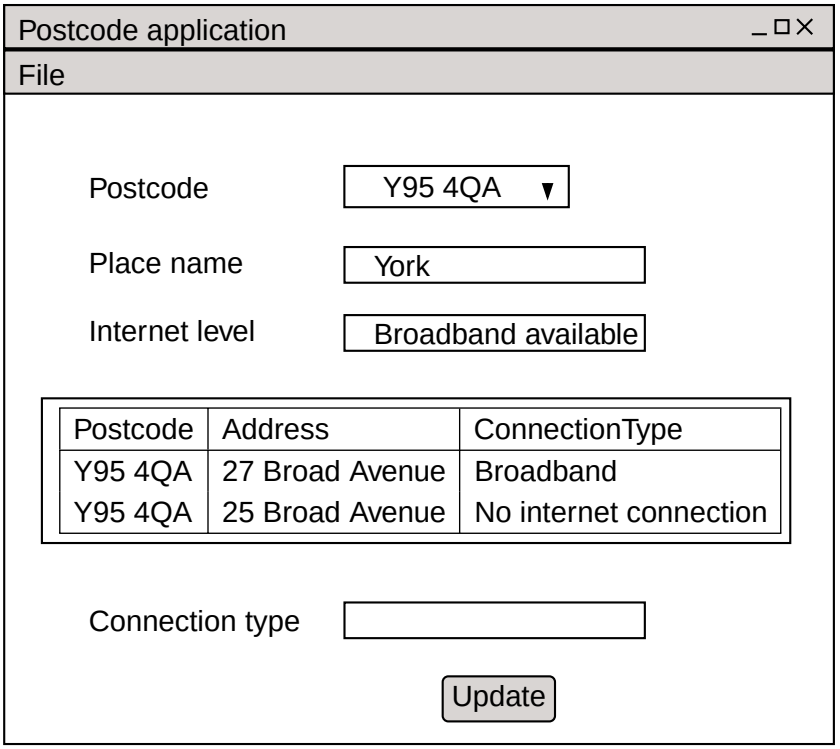
Project specification: Postcode application

The application is intended to allow the user to view and make some edits to the postcode database. The application should possess:

- a way to view the place name and internet level at a given postcode;
- a way to edit the internet level at a given postcode;
- a way to view the data about individual addresses at a given postcode;
- a way to edit the connection type at a given address.

The application should not allow any data to be edited other than that specified above.

A sketch of the graphical user interface and a partially completed property table are given below.



Property table: Postcode application

	Name	Property	Initial value
Containers			
Form	MainForm	Text	Postcode application
MenuStrip	mainMenu		
Controls			
Label	postcodeLabel	Text	Postcode
comboBox	postcodeComboBox		
Label	placeNameLabel	Text	Place name
TextBox	placeNameTextBox		
Label	internetLabel	Text	Internet level
TextBox	internetTextBox		
DataGridView	residenceDataGridView		
Label	connectionLabel	Text	Connection type
TextBox	connectionTextBox		
Button	updateButton	Text	Update

Property table: Postcode application – menu bar

	Name	Property	Initial value
Container			
MenuStrip	mainMenu		
Controls			
Menu	fileMenu	Text	File
MenuItem	saveMenuItem	Text	Save
MenuItem	exitMenuItem	Text	Exit

(a) Draw up a property table for any non-visual components. The name of the data set should be postcodeDataDataSet. Note that the displayed addresses should be those for the postcode selected in postcodeComboBox. [*5*]

(b) (i) Write down an extension to the main property table shown above that gives details of the data bindings that can be set up during design time.

(ii) Describe how you would restrict a user's editing permissions for controls on the main form in order to meet the given specification. [*6*]

(c) (i) Write the design code body for the OnClick event handler of the *Save* menu item. There is no need to include a comment.

(ii) Write the design code body for the OnClick event handler of the *Update* button. There is no need to include a comment.

(iii) How would you ensure that an attempt is made to save updated data to the underlying database when the application is closed? [*7*]

[END OF QUESTION PAPER]