COMP1022Q Review Questions Week 4

These practice questions, as well as the answers, are released for the content of each week. These are just for you to practice. You don't have to hand them in. There's no marks for doing them, but it will help your understanding if you go through them.

Q1)

Here is some VBA code.

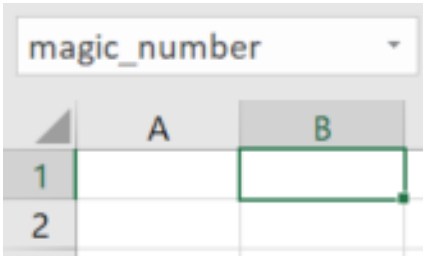
```
Sub CheckSurname()

    Dim Name As String

    Name = InputBox("Please enter your first name:")
    Name = "ROSSITER " & Name
    MsgBox "If you marry Rossiter then your name will be this"
    MsgBox "Using the Chinese style: " & Name & "!"

End Sub
```

Somebody with the first name 'Fluffy' enters their name in the input box and presses Enter. Two message boxes will be shown, one after the other. What will be the text shown in the **last** message box?

Answer: _____

Q2)

Here is part of a worksheet.



In this situation, which one of the following pieces of VBA code **does not** refer to cell **B1**?

A) `Cells(2, 1).Value`

B) `Range("magic_number").Value`

C) `Range("B1").Value`

Answer (A/B/C): _____

Q3)

A student has written a cell formula to give feedback using words, according to the midterm score he/she receives. The following cell formula has been entered into cell B2.

```
=IF( B1>85, "Excellent", IF(B1>65, "Good",
                         IF(B1>50, "Fair", "Poor")) )
```

Now the student decides to convert this cell formula into VBA code. You need to complete the following VBA code so it behaves the same way as the above formula, which is in cell B2. Write the missing code in the blanks. You cannot amend any of the VBA code given to you. You can assume a midterm score is always an integer.

```
Sub Compare()

    Dim Score As Integer
    Score = Range("B1").Value


    If _____________________________ Then

        Range("B2").Value = "Excellent"

    ElseIf _____________________________ Then

        Range("B2").Value = "Good"

    ElseIf _____________________________ Then

        Range("B2").Value = "Fair"


    ____________________

        Range("B2").Value = "Poor"
    End If

End Sub
```

Q4)

Here is some VBA code.

```
Sub CheckAge()
    Dim Age As Integer
    Age = InputBox("How old are you?")

    If Age < 18 Then
        If Age < 0 Then
            MsgBox "It is not possible!"
        Else
            MsgBox "You are still a kid."
        End If
    Else
        MsgBox "Hey, big boy!"
    End If
End Sub
```

You need to fill in the blanks so that the following subroutine behaves the same way as the one shown above. You cannot add any extra code. You can only fill in the missing code in the two blanks.

```
Sub CheckAge()
    Dim Age As Integer
    Age = InputBox("How old are you?")


    If _____________________________ Then

        MsgBox "Hey, big boy!"


    ElseIf _____________________________ Then

        MsgBox "It is not possible!"
    Else
        MsgBox "You are still a kid."
    End If
End Sub
```

Q5)

Here is a worksheet.

	A	B
1	**Venue**	**Capacity**
2	Lecture Theatre C	213
3	Room 2502	122
4	Lecture Theatre A	404
5	Room 2407	126
6	Lecture Theatre G	135

Here is a VBA macro.

```
Sub FindVenue()
    Dim Row As Integer
    Row = 2
    While Cells(Row, 2).Value <= 200 And Row < 6
        Row = Row + 1
    Wend
    MsgBox Cells(Row, 1).Value & _
    " is the first venue in the list with capacity larger than 200."
End Sub
```

You know what the VBA code `And` does, because it behaves the same as cell formula AND() which we experienced before. `_` means 'the code is continued below'.

If you run the above code, what is the venue shown in the message box?

Answer:

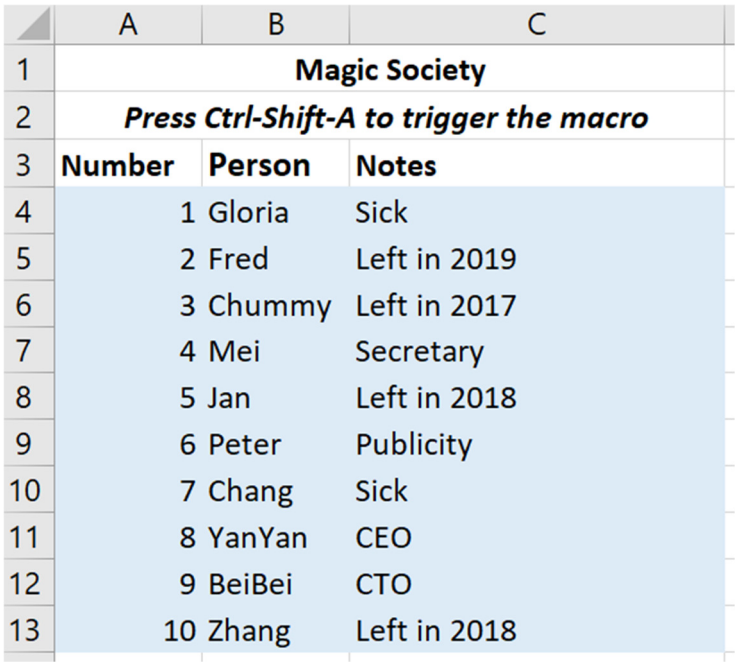
_____ is the first venue in the list with capacity larger than 200.

Q6) For this question you need to fill in the blanks in the following VBA subroutine so that at the end of the subroutine the variable `MyNumber` contains **a random even number** between 10 and 50 inclusive (i.e., the variable will contain one of 10, 12, 14, 16, …, 48, 50). Don't forget that `Rnd()` generates a number from 0 up to 1, including 0 but not including 1.

```
Sub GenerateRandomEvenNumber()
    Dim MyNumber As Integer

    MyNumber = Int(Rnd() * _____________ ) + _____________

    MyNumber = MyNumber * _____________
End Sub
```

Q7)

Here is a worksheet.



	A	B	C
1	**Magic Society**		
2	***Press Ctrl-Shift-A to trigger the macro***		
3	**Number**	**Person**	**Notes**
4	1	Gloria	Sick
5	2	Fred	Left in 2019
6	3	Chummy	Left in 2017
7	4	Mei	Secretary
8	5	Jan	Left in 2018
9	6	Peter	Publicity
10	7	Chang	Sick
11	8	YanYan	CEO
12	9	BeiBei	CTO
13	10	Zhang	Left in 2018

Here is a VBA macro.

```
Sub CalculateTotal()
    Dim Total As Integer
    Dim Row As Integer
    Dim Col As Integer

    Total = 0
    Row = 4
    Col = 3

    While Row <= 13
        ThisNote = Cells(Row, Col).Value

        If Left(ThisNote, 7) <> "Left in" And _
           ThisNote <> "Sick" Then
            Total = Total + 1
        End If
        Row = Row + 1
    Wend

    MsgBox "Now there are " & Total & " people in the magic society"
End Sub
```

You know what the VBA command `Left()` does because it works the same as the cell formula command `LEFT()` which we experienced before on the course. For example, the cell formula `=LEFT("Goodbye", 4)` produces `"Good"` and the VBA code `Left("Goodbye", 4)` also produces `"Good"`. Similarly, you know what the VBA command `And` does, because it behaves the same as cell formula `AND()` which we experienced before.

When the macro shown above is executed, a message box appears which says how many people are in the magic society. What is the number shown in the message?

Answer: _____

Q8)

Here is a worksheet.

	A	B	C
1	**Month**	**Profit**	
2	Jan	$1,000	
3	Feb	$2,600	
4	Mar	-$800	
5	Apr	$3,700	
6	May	$2,700	
7	Jun	-$700	
8	Jul	-$280	
9	Aug	$3,250	
10	Sep	-$280	
11	Oct	-$500	
12	Nov	$2,600	
13	Dec	$3,100	
14			

Cell B2 to cell B13 have been displayed using the currency format. You can ignore the dollar sign shown at the start of those cells when you answer this question, because the dollar signs are not actually inside the content of the cells, they are just displayed by Excel.

Part A)

If you run the following VBA subroutine, what is the number shown in the message box?

```
Sub PartA()
    Dim Row As Integer
    Row = 2
    Do While Cells(Row, 2).Value >= 0 And Row <= 13
        Row = Row + 1
    Loop
    MsgBox Row
End Sub
```

Answer: _____

Part B)

If you run the following VBA subroutine, what is the number shown in the message box?

```
Sub PartB()
    Dim Row As Integer
    Dim Count As Integer
    Row = 2
    Count = 0
    Do While Row <= 13
        If Cells(Row, 2).Value >= 0 Then
            Count = Count + 1
        End If
        Row = Row + 1
    Loop
    MsgBox Count
End Sub
```

Answer: _____

Answers to Week 4 Review Questions

Q1)
The answer is:
```
Using the Chinese style: ROSSITER Fluffy!
```

Let's look at the code line by line.

```
Dim Name As String
```

This first line of code makes a variable called Name which will hold text. In the world of programming *string* means *text*.

```
Name = InputBox("Please enter your first name:")
```

The second line of code takes the input from the user and puts it into the variable. In this case the user enters `Fluffy` so `Fluffy` goes into the variable.

```
Name = "ROSSITER " & Name
```

In the third line of code, two things are concatenated (=stuck together). The first thing is `"ROSSITER "`, which has a space at the end. The second thing is the name entered by the user, which is `"Fluffy"`. The result of sticking the two pieces of text together is `"ROSSITER Fluffy"`. This result goes back into the variable Name. After this line of code the variable Name contains `"ROSSITER Fluffy".`

```
MsgBox "If you marry Rossiter then your name will be this"
```

The fourth line of code shows a simple message on the screen. When the user presses 'OK' the message will disappear.

```
MsgBox "Using the Chinese style: " & Name & "!"
```

In the last line of code a second message is displayed. The message is a concatenation of three things. The first thing is `"Using the Chinese style: "`, which has a space at the end. The second thing is the text stored in the variable Name, which is `"ROSSITER Fluffy"`. And the third thing is an exclamation mark, `"!"`. When they are all put together this result is shown in a message box:

```
Using the Chinese style: ROSSITER Fluffy!
```

Q2) The answer is A.
In the situation shown in the worksheet `Range("magic_number").Value` and `Range("B1").Value` both refer to cell B1. However, `Cells(2, 1).Value` means 'the second row, the first column', which is cell A2 and not cell B1. So the answer is A.

Q3) The answers are shown underlined below.

```
Sub Compare()
    Dim Score As Integer
    Score = Range("B1").Value

    If Score > 85 Then
        Range("B2").Value = "Excellent"
    ElseIf Score > 65 Then
        Range("B2").Value = "Good"
    ElseIf Score > 50 Then
        Range("B2").Value = "Fair"
    Else
        Range("B2").Value = "Poor"
    End If
End Sub
```

Q4) The answers are shown underlined below.

```
Sub CheckAge()
    Dim Age As Integer
    Age = InputBox("How old are you?")

    If Age >= 18 Then
        MsgBox "Hey, big boy!"
    ElseIf Age < 0 Then
        MsgBox "It is not possible!"
    Else
        MsgBox "You are still a kid."
    End If
End Sub
```

Q5) The answer is shown underlined below.

Lecture Theatre C is the first venue in the list with capacity larger than 200.

From `Row = 2` you know that the value of the variable `Row` starts with the value 2.
The top of the loop code looks like this:

```
While Cells(Row, 2).Value <= 200 And Row < 6
```

The 2 inside `Cells(Row, 2).Value` means the loop will consider the cells in the second column (column B). The second row and the second column means cell B2. Cell B2 has a value larger than 200, as you can see in the image below.

	A	B
1	**Venue**	**Capacity**
2	Lecture Theatre C	213

That means the condition of the `Do While … Loop` is false. That means the loop ends. So in this Part A code, the content of the loop never gets executed. The value in the first column of row 2 (cell A2) is shown in the message box, so we see this message:

Lecture Theatre C is the first venue in the list with capacity larger than 200.

Q6) The answers are shown underlined below.

```
Sub GenerateRandomEvenNumber()
    Dim MyNumber As Integer

    MyNumber = Int( Rnd() * 21 ) + 5

    MyNumber = MyNumber * 2
End Sub
```

The target is to make a random number which is one of 10, 12, 14, 16, …, 48, 50.

So we need to generate a random even number. One way to do that is to generate a random integer and multiply it by 2. That means that the random integer generated should be half of the range of the even number before it gets doubled, i.e. in this case it should be in the range 10/2 to 50/2. Then we can can double it to make a random number in the range 10 to 50.

Considering all the steps, to generate a random integer between 5 to 25, we can do this:

1. Generate a random number between 0 to 0.99999 using `Rnd()`.
   The range of the random number is [0, 1).
2. Now we multiply the number by 21. The range is now [0, 21) , with a decimal place.
3. Now we truncate the number using `Int()`. The range is now [0, 20] , just integers.
4. Now we add 5 to the number. The range is now [5, 25] , just integers.
5. Finally, we can double the number, to make a range of [10, 50] , just integers.

This is the approach taken in the code shown above.

Q7) The answer is 4.

The loop looks like this:

```
Row = 4
While Row <= 13
    ThisNote = Cells(Row, Col).Value
    If Left(ThisNote, 7) <> "Left in" And _
        ThisNote <> "Sick" Then
        Total = Total + 1
    End If
    Row = Row + 1
Wend
```

The value of the variable row changes from 4 to 13 inclusive. So that means each cell from cell C4 to C13 inclusive is considered by the loop. For each of those cells the code does a test: if the left side of the cell contents does not contain "Left in" **and** the cell is not the same as "Sick" then we increase the total count by 1. In other words, if the person has not already left the society ("Left in 2018", "Left in 2019", and so on) and the person is not ill ("Sick") then the person is regarded as still being in the society, so we increase the count by 1. There are 4 people who have not left and are not sick, so the answer is 4.

Q8)

Part A)
The answer is:
4

In the VBA code, the condition part of the `Do While … Loop` is `Cells(Row, 2).Value >= 0 And Row <= 13`. This means we look at the cell value at row number `Row` and column number 2 (i.e. column B). While the cell value is larger or equal to 0 and its row number is less or equal to 13, we continue to execute the loop body repeatedly. The loop body simply increases the value of `Row` by 1. As a result, starting from row 2, we look at the cell values in each row in column B one by one until the cell value is negative (i.e. < 0). Since cell B4 is negative (-800), the loop ends. The value of `Row` is 4.

Part B)
The answer is:
7

In the VBA code, the `Do While … Loop` examines each cell in column B between row number 2 to row number 13. In every iteration, the `If` statement looks at the cell value. If the cell value is larger or equal to 0, it increases the value of the variable `Count` by 1. Therefore, the VBA code counts the number of positive values among cell B2 to cell B13. There are 7 positive values.