(Part II) Lab-session 2

1) Record a Macro, which when activated draws a "rainbow" into the cells A1:K6 of your worksheet. Call your Macro "Rainbow" and design it such that it can be activated with Ctrl + r.

· In order to produce the Macro, when recording, shade the row A1:K1 with "Fill Color" Red, A2:K2 with the "Fill Color" Orange, etc. (seven colours in total)

· Once you have carried out the recording view the VBA code of the recording you have produced and try to figure out what it does. Once you have understood what the code is doing, creating a new macro called "bowrain" by editing the code in such a way, that the colours in the rainbow will appear in reversed order, that is A1:K1 should be shaded with Violet, A2:K2 with Blue,... and A6:K6 with Red.

2) Record a Macro which copies the content of the cells A1:G6 to the cells G21:M26. Give your macro a meaningful name and design it such that it can be activated with Ctrl + c. Test this Macro together with the Macro "Rainbow" and see what happens.

3) Create three customized buttons for the three Macros recorded in tasks 1 and 2. Write the Macros names onto them and test them in various different orders.

4) Record a Macro similar to the one we saw in the lecture, which when activated computes the sum of the column A1:A25 of your worksheet. Call your Macro "SumA1A25" and design it such that it can be activated with Ctrl + s. Use this Macro to compute the sums

$$\sum_{a=1}^{25} 2a \qquad \sum_{a=11}^{35} a \qquad \sum_{a=1}^{25} 2^a.$$

To fill in the values of cells A1:A25 for the last sum you will need to use something a bit more advanced than the Autofill function. First enter the first two values you want to sum in cells $A1$ and $A2$ (that is 2 and 4). Then select the range A1:A25 and in the Home Tab select the "Fill" option. In the drop down menu choose "Series". In the window that will open you need to choose the following options: "columns", then "growth" and finally "trend".

5) For those of you who already finished exercises 1)-4), something a little more challenging. Have a look at one extra example of the use of loops that is now available from the module web-page[1]. Once you have understood that example you can try to do the following problem: consider the function $f(x, a) = ax(1 - x)$. Write down a code for a <u>subroutine</u> (not a UDF!) that does the following:

- The subroutine reads the values of x and a from cells $A1$ and $A2$ of the Excel worksheet.
- When run it will write the value of the n-folded composition of the function with itself in cell $A3$. For this it will read the value of n from cell $B1$ in the Excel worksheet.
- For example, for $n = 2$ we want to compute $f(f(x, a), a)$, for $n = 3$ it will be $f(f(f(x, a), a), a)$. So every time the program executes the loop the new value of x becomes the last value of $f(x, a)$ and a is fixed from the beginning. This is basically the same loop as in the notes, but with one more variable "a".

In order to make your macro read data from the Excel worksheet and write data in the Excel worksheet you need to know the following. The code line:

Range("A1").Value=3

writes the value 3 into the cell $A1$.

The code line

x=Range("A1").Value

defines a variable x whose value inside the programme is the value inside cell "A1" of the Excel worksheet.

Once you have made your subroutine work, you can check the following:

- If you take a value of $1 < a < 3$ and $x = 0.5$ you should find that for values of n sufficiently large (say 15 or more) the value of the n-fold composition of the function does not depend too much on n. (Take $n = 15, 17, 30$ and you should see that you always get more or less the same number).
- If you now take a slightly above 3 (say 3.2) and $x = 0.5$ you should observe that for values of n sufficiently large (say 15 again) the n-fold composition oscillates between two values (that is for n=15 you get a value, for n=16 you get another value, for n=17 you get back the same value as for n=15, for n=18 you get back the same value as for n=16 etc.)
- Finally, if you take the value $a = 0.39$ and $x = 0.5$ you should see that even for values of n rather large, the n-fold composition gives always entirely different values as you do one more iteration (e.g. the value for n=15 is entirely different from n=16 and this is different from n=17 etc.)

The equation $f(x) = ax(1 - x)$ is known as <u>logistic equation or logistic map</u> and is a very famous (and simple) example of a system that exhibits chaotic behaviour for values of "a" close to 4. The equation was originally used to describe the evolution of a population, where x is the number of individuals in the population for a given year and $f(x)$ represents the number of individuals one year later. If you "google" logistic equation in the web you will find lots of web sites discussing it!

[1] http://www.staff.city.ac.uk/o.castro-alvaredo/PROGRAMMING/example-loops.pdf