PRACTICAL WORK 1

I would recommend working on these questions in sequence after you have listened to the lecture recordings, and we will do some in class. The once we do in class are particularly useful to support the material in the lectures, the others are meant to take you beyond the lecture material. The numbers in brackets before the question refer to the lecture part that the questions relate to. Questions marked with a * go deeper into the mathematical background, and are there to deepen your understanding. I do not expect everyone to be able to do these.

Installing and running XPP on your computer under Linux (see Appendix for Windows)

The XPP programme is freeware, it is free to use. To run it under Linux you will first need to install `xpp` and `xppaut` in the terminal:

```
sudo apt-get install xppaut
```

To run XPP all you need to do type `xppaut` in the terminal. A GUI will open. Before doing so, please `cd` to where the `.ode` files are stored (ideally, in your weekly coursework directory, under the `code` directory). We can now start

(1-2) Make a bifurcation diagram for the continuous logistic equation.
The purpose of this exercise is to get familiar with XPP and AUTO. We will recreate the bifurcation diagram that we had analytically constructed for the logistic equation. First we need to tell XPP what the model is, and what parameters to use. This is done by creating a file in which this is all specified. I have made one for this model, it is called `logistic.ode.`It contains the following lines of code:

```
# continuous time logistic model

# first define the parameters
par k=3, r=1

# now the model
n' = r*n*(k-n)

# and the initial conditions
init n=0.01

# the following lines define the outlay of some of the
# XPP and AUTO screens to save you the hassle of doing this yourself

@ ylo=0, yhi=5, xhi=3, total=3
@ autoxmin=-5, autoxmax=5, autoymin=-5,autoymax=5, parmax=5, parmin=-5

d
```

This script should already be in your `Code` directory in a file called `logistic.ode`. In the terminal, `cd` to the `Code` directory, and launch `xppaut`. You should see the different `.ode` files. Select `logistic.ode`, which will open a new blank plot.

Now, before you can run the model you will need to **set the parameters and the initial conditions** to something that works for you. To see the initial conditions click on the little box "*Ics*" in the top of the GUI window. Change the initial conditions if you want to, then click OK and close the little window for the Ics. With the other boxes in this line you can set the parameters (*Param*) and look at the equations for the model (*Eqns*).
You can set the axes for the display by clicking *ViewAxes*, and in this case it makes sense to use 2D. You can set the variables you plot, and the ranges over which you plot.

Once you have done all this, you are ready to run your model. In this example there is no real need to change anything, as all settings are suitable for this problem (that is what the lines in the ode file starting with @ do). However, should you want to make your own `ode` files then this is something you will need to set yourself.

To **run the model**, click on *Initialconds* and select *(G)o*. Alternatively, use the keystroke commands "I" followed by "G". The solution to the differential equation will appear on your screen. You can run a solution with new initial conditions by clicking on *(N)ew*, and typing in the new initial conditions. If you want to extend the solution you have already made for longer time, select *(L)ast.*

To see what the solution looks like for a range of solutions, click *Initialconds* and select *Range* or use the keystroke commands "I" followed by "R". A window pops up in which you can specify the range of initial conditions. Investigate what happens

(1) For a range of different initial conditions, ranging from 0 to 3.
(2) For a range of values of the parameter k, ranging from 0 to 5
(3) For a range of values of the parameter r, ranging from 0 to 5.

Next, we will **make a bifurcation diagram** for the logistic model. The first thing to do is to find an equilibrium to start the diagram from. Click on *Initialconds* and select *(G)o*, followed by selecting *(L)ast* until you can see the solution has reached equilibrium (or quicker, type I G, then I L repeatedly).

Now that you have found an equilibrium point we will take this point to AUTO. Select *File* and then *Auto*. A new window will open. Before you can continue the equilibrium and follow it under a change of parameters, check first that the display shows what you want it to show. For this exercise we want a plot of N vs k, and in the range -5 to 5 for both parameters. Check if this is correct by first clicking on *Parameter*, here k should be parameter 1. Next click on *Axes*, go to *hI-lo* and check that you plot N vs the parameter k, and that the boundaries are -5 and 5. You also need to check over what range the program will calculate the diagram. You do this by going to *Numerics* and check *Par Max* and *Par Min*. Once you have done that you can plot the bifurcation diagram. Click on *Run* and then on *Steady state* (indicating you are dealing with an equilibrium). A red line should appear. You have now drawn the bifurcation diagram from you starting point (k=3) to your end point (k=5). The diagram is still incomplete because you have no values below k=3.

To complete the diagram, you need to do the calculation backward. Go to *Numerics*, change the value for *Ds* from 0.02 to -0.02. Click *Ok* and in the AUTO window click *Grab*, this allows you to access the results you have previously computed. A cross will appear in your diagram indicating which point you have selected. By using the arrows and tabs you can step through your solution, following the line in the diagram. Here you want to select the first point. You do this by hitting the return immediately after you have clicked *Grab*. The big cross will disappear. Now click *Run*, the rest of the diagram will appear.

Use Grab and the arrows to walk through your solution. Note the little circle at the bottom corner. It displays the Exponential values of the eigenvalues (stable within the circle, unstable outside). Note that at the crossing point of the lines the text line at the bottom shows the word BP. This is a special point that AUTO flags up.

To stop and close your windows: *Close* the Auto window. To close the XPP window type *File Quit Yes*. You will probably use this frequently, and it can also be done with the keystrokes If *F Q Y*. You might want to memorise this.

(1-3) Levins' metapopulation model with habitat destruction is given by the equation:

$$\frac{dp}{dt} = mp(1 - D - p) - ep$$

What is the equilibrium? Is the equilibrium always stable?
First try to do this with pen and paper.

Use XPP to plot dynamics. Plot the solutions for a range of values of D. can you plot the bifurcation diagram in m?
You can use the file `metapop1.ode` if you like. Note: I did not set the parameter values or define the axes as I did for the logistic growth model, so you will have to do this yourself.

(1-3) The Levins' metapopulation model with rescue effect is given by

$$\frac{dp}{dt} = mp(1 - p) - e_0 p \exp[-ap]$$

Can you solve for the equilibrium using pen and paper? Why is this difficult? Can you plot the equilibrium and show how it changes under a change of the parameter m? (This is possible but you will need to think outside the box. Hint: swap the x-axis and y-axis over). Can you work out the stability of the equilibrium? (*This is a difficult part of the exercise.)

If you succeeded in plotting the equilibrium by hand, well done! If not, not to worry, as this is exactly what computer programmes are good at. We will next use XPP to generate the equilibrium numerically. Use XPP to plot the equilibrium as a function of m. You can use the file `metapop2.ode`. What is happening at the point where the red line stops? Investigate the population dynamics around that point by going back to XPP. Plot the dynamics of the model for a range of initial conditions. You can do this by using the *(R)ange* option under *ICs.* Use a range of initial conditions from 0 to 1 for parameters m=0.8, a=3, e=1. What do you see? How does this relate to the bifurcation diagram you made earlier?

(1-1)* Work out the eventual multiplication factor (growth rate) for the Fibonacci sequence. The answer is known as the Golden Ratio. One way is to rewrite Fibonacci's model as a recurrence or in difference equations and distinguish between juvenile rabbits, which do not reproduce yet, and adults, who do reproduce. This is easiest done as a matrix model.

(1-1)* What happens if Fibonacci's rabbits die after their second year? Can you write this as a recurrence or difference equation?

(1-1) Calculate the per capita growth rate of the human population. We have done this in the lecture, but I think it is important to get this basic calculation, so here it is once more. In the year 500 the human population had a approximate size of 190 million people. In the year 900 the approximate population size was 240 million. On the basis of these figures calculate the *per capita* growth rate per year. Do this under assumption that the population grows exponentially and use the appropriate formula.What is the doubling time of the population over the period above? (For reference and contrast, the current doubling time of the human population is about 35 years.

(1-1)* Show mathematically that when plotted on semi logarithmic scale the continuous time exponential growth curve follows a straight line. This is straightforward, but it is insightful to do this for yourself.

Appendix: installing and running XPP under Windows

To run XPP under Windows you will need an X server programme. Xming is a freeware X server. You can find it by either googling Xming, or going directly to http://sourceforge.net/projects/xming/. Click on Download (the green button) and save the file on your computer. Once you have downloaded it, click on the `.exe` file to install the programme. Unless you want extras, do not change any of the suggested settings. If it is installed correctly, you should see an entry Xming under “All Programs” (or equivalent).

Next, install XPP. You do this by going to the XPP page http://www.math.pitt.edu/~bard/xpp/xpp.html. Click on “Easy windows”, and on the next page on “XPP installation”. This will download the file “xppwin.zip” on your computer (or do it directly by going to http://www.math.pitt.edu/~bard/bardware/binary/latest/xppwin.zip). Once you have downloaded it, open the zip file (right click and “Open with” and choose “Compressed (zipped) folders”).

You should now copy the folder “xppall” to the root of your C drive (into C:\). (If you copy it somewhere else it will not work correctly unless make changes to the .bat file that we need later on.) Once you have copied the xppall folder in your C drive, open it. In the folder you will find the file `xpp.bat`. Make a shortcut to this file on your desktop, by rightclicking xpp.bat and “create shortcut”. Copy the shortcut to your desktop.

To edit your `.ode` files it is convenient to display file extensions. To do so, in Windows Explorer:

- Click **Folder and search options**.
- Click the **View** tab.
- find **Hide extensions for known file types**, un-check this line by clicking the check box
- Click **OK**

To run XPP you first need to start you Xserver. Go to “All programs”. Go to “Xming” and select “XLaunch”. There is no need to change any of the options unless you want to. Once you have done this you will see the Xming icon on your taskbar. To run XPP all you need to do is to select on .ode file, and these can be found in the “ode” folder in “xppall” and drop it on the shortcut to `xpp.bat`. After a short while this should start up the main window of XPP.
