Name: Abdullah Shahzad
Course: CS 152
Section: B
Date: 10/20/2021

Project 5 Report:

Title: *Simulating Models to Control The Population of Elephants in Kruger National Park*

Abstract:
This project was about controlling the population of elephants in Kruger National Park. We have two methods to control the population of elephants: we can either dart female elephants with a contraceptive to terminate and prevent pregnancies for 22 months after darting, or we can cull the elephant population at random to keep it less than the maximum amount that the park can hold.
My program simulates the growth of the population of elephants, taking into account elephants that die at any stage of their life, female elephants that get pregnant, and new elephants that are born. The program then asks the user for a command-line argument that decides whether the method for controlling the population is darting or culling. Details about the command-line argument are mentioned in the usage statement but generally, if the user enters 0 in the command line, culling is adopted, and if the user enters a number between 0.0 and 1.0, darting is adopted.
The program outputs a multidimensional list. A list is a way of representing data; it is surrounded by square brackets and keeps objects, separated by commas, inside. A list inside a list is basically a set of objects inside square brackets surrounded by another pair of square brackets, making the list that is inside the list an object.

Example of a list = [1, 2, 3]
Example of a multidimensional list = [[1, 2, 3], [1, 2, 3]]

The output of this project is a list containing lists of statistics for each year, the total number of elephants, the average number of calves, adult males and females, and the amount of elephants that are culled if culling is opted for (otherwise this number will be 0).
Through this project, we can determine how much culling is required on average and what percentage of the population should be darted to keep it controlled.

Results:

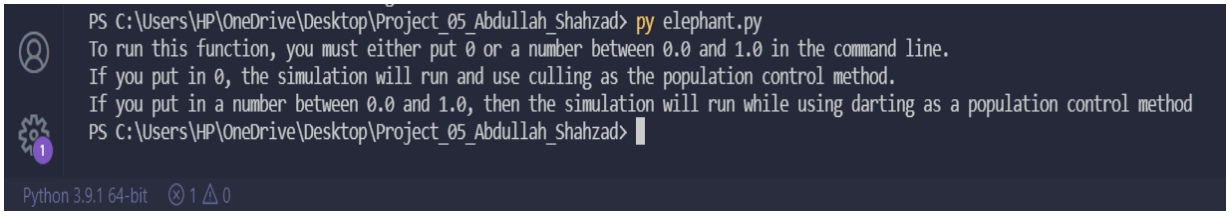


Fig 1.1

Fig 1.1 shows the usage statement that would be printed to the terminal in case the user did not enter an appropriate command-line argument

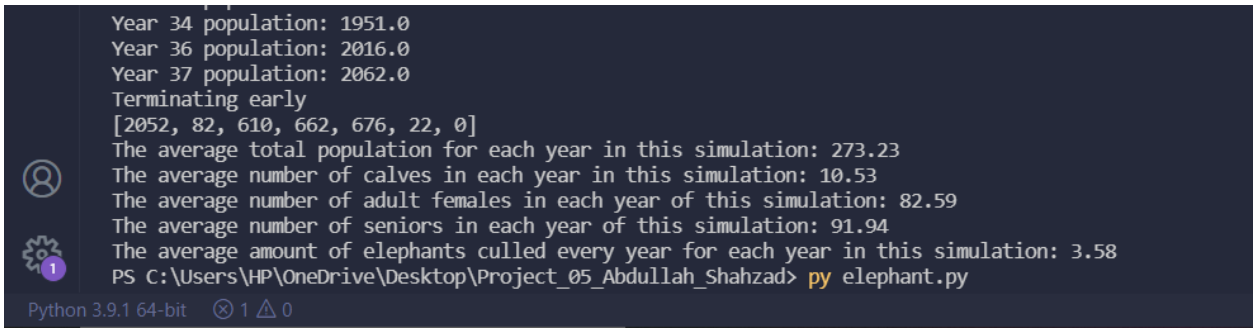


Fig 1.2

The maximum capacity entered for Kruger National Park in Fig 1.2 is 1000

Figure 1.2 shows the results that the user would get if they called this program with the command line argument 0.3. The function terminates itself after year 37 because the population of elephants exceeds twice the maximum capacity.

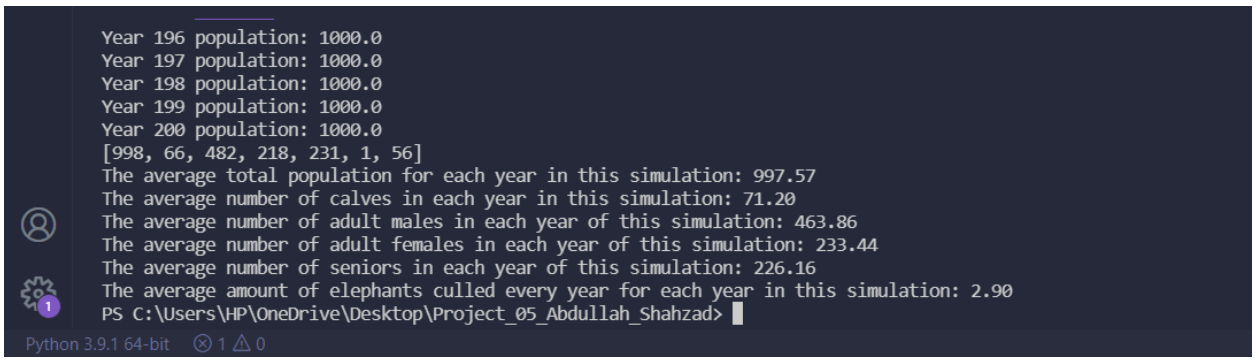


Fig 1.3

In Fig 1.3, we see the results that the user would get when they enter the command-line argument 0.0 (opting for culling instead of darting) and the maximum capacity of the park is 1000. The function controls the cullign to execute all 200 years, maintaining the population at the maximum after the maximum is reached organically.

The maximum capacity value can be changed, and the darting percentage can be changed to simulate the population again and again and see what happens in each simulation.

Reflection:

Concepts I utilized in this project were list-slicing methods, modular design, making multidimensional lists, and assigning indexes to variables. All of these made coding easier to follow and then also easier to debug because I'd know where to look if anything went wrong and I could check pieces of code in isolation.
Real world applications of this project, besides simulating animal populations, can be used to monitor human population growth, the lifespan of objects (e.g. cars by using variables like amount of accidents, distance travelled etc.), and can also be used to check how different variables impact the overall simulation (like dartingg or culling in this simulation).

Follow-up Questions

1. What is the difference between a tuple and a list?

A tuple has parenthesis () around it while a list has square brackets around it []. A tuple is immutable while a list is mutable. A tuple also takes up less space than a list so it's easier to compute.

2. Why is it helpful to use a variable like `IDXNumYears` instead of a number (e.g., 3) when accessing a particular element of a list?

When accessing multidimensional lists, it becomes hard to remember each index number, especially if we have a bunch of different multidimensional lists. That is why, labelling the index number in a human readable way makes it easier for us to index into them

3. Where might you anticipate errors might be made in the simulation that would impact the results? How would ensure the final simulation model would produce reliable results to support well-informed wildlife conservation decisions about this elephant population?

It seemed as if the random library changed from one python version to another, making results from a particular model of python predictable and similar, this may suggest that the results aren't as random or all encompassing as we might like. Besides this, while deciding when an elephant gets pregnant, a random number between 1 and 22 is assigned instead of 1 (which would be where a pregnancy starts), suggesting that there are unrealistic leaps of time because obviously an elephant wouldn't immediately become 20 months pregnant.

Extension (optional):

Extension 1:

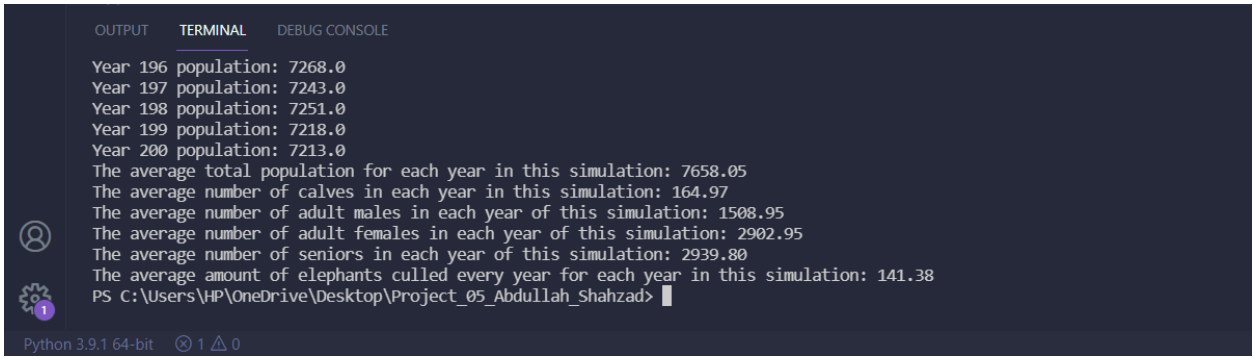
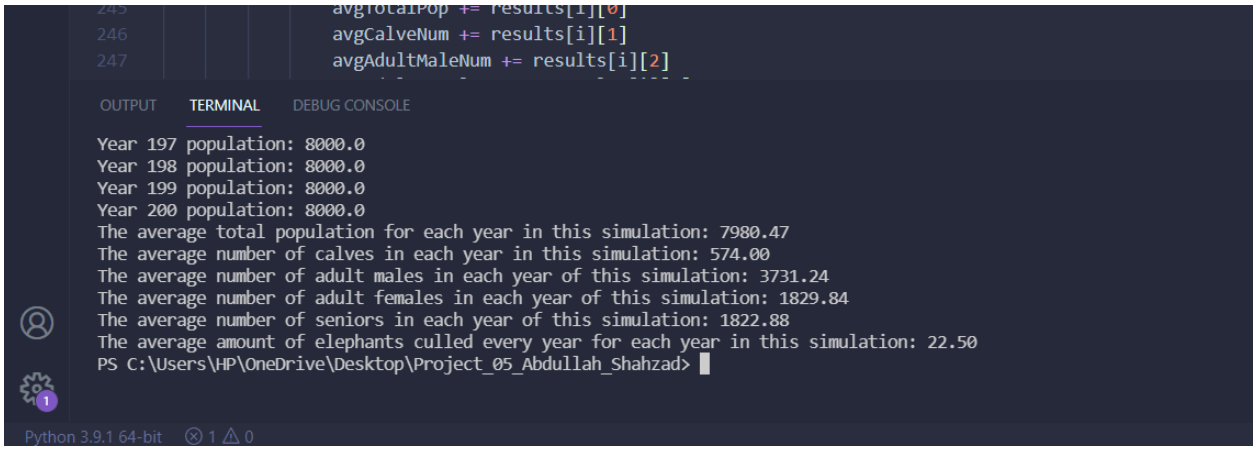


Fig 2.1

In Fig 2.1, the darting probability used is 0.457. As a result, a stable population is obtained. This means that if the national park rangers dart 457 out of every 1000 female elephants, the park population can be maintained at just less than 8000 elephants.

Extension 2:



After comparing the results for using cullign as a population control method and darting as a population control method, the statistics obtained are drastically different. In Fig 2.1, you can see that darting leads to a lesser number of average calves while culling leads to a lot more calves surviving. Note that right now, the elephants culled are random, which means that realistically, rangers can choose which elephant to cull and not to cull. This means that instead of culling calves they can cull elephants that are sick/old. On the other hand, if we're darting elephants, it will obviously be the calves that just aren't born meaning that rangers have a lot less autonomy over what elephants they get to keep and that might not exactly be ideal for tourism either.

Extension 3:

The system is extremely sensitive to changes in darting percentage. I had to go into 3 decimal places to get an appropriate darting percentage to keep the population stable. If the darting probability went even slightly above 0.5, the amount of elephants left would decrease too drastically. This also means that when culling, rangers have a lot more

control over how many elephants they want compared to darting all of them based on a particular probability. You can see for yourself in Fig 2.2 that the population is almost stable at 8000.

Acknowledgements: Professor Allen Harper