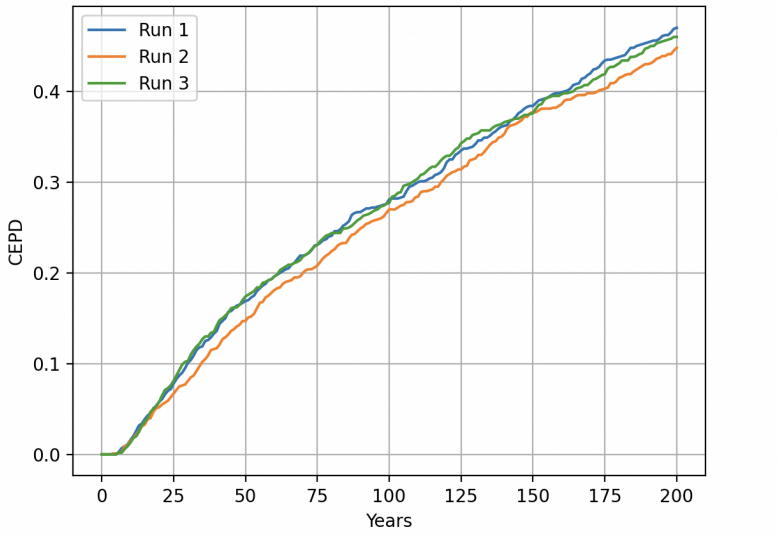
Svita Kiran
Project 04: Penguin Population Viability Analysis (PVA)

Abstract
The key concepts of this program were to develop simulation functions and use randomization. What this program is meant to do is to model the change in the population of penguins over time, specifically 201 years, given ecological pressures which in this case is an El Nino cycle of either 3, 5, or 7 years. The program has different factors such as the population size, female penguin probability, El Nino probability, population growth rate in an El Nino or non- El Nino year, maximum carrying capacity, and the minimum viable population. All these variables have default values but the user can select the number of simulations to run and the number of years between an El Nino event which influences the outcome. These inputs cause an output of the probability of penguin survival and the CEPD (cumulative extinction probability distribution) which is the cumulative extinction probability of each year simulated.

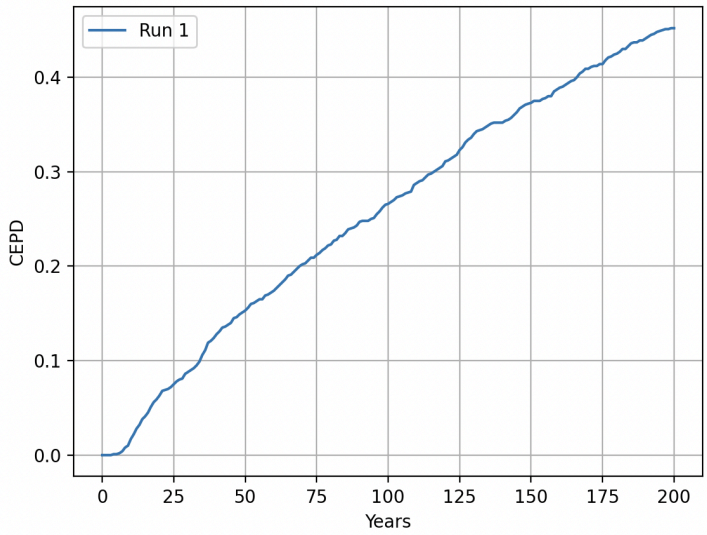
Methods
In this project, one place I used computational thinking was when simulating a year. Another place I used it was with randomization because in the code, it helped to simulate the real world better because of its unpredictability.

An algorithmic feature that was essential to the program was iteration and slowly building up off of each function because it helped simplify the program a little bit. Using this, I was able to run the program each time I made a small change and see the exact changes I made in the output. For example, I started by simulating a year, and eventually I was able to build off of that function exactly to simulate 201 years. This also helped me debug my problems because I originally had many errors in the results, and instead of going through the whole code step by step, I knew which general part of the code I had gone wrong with.

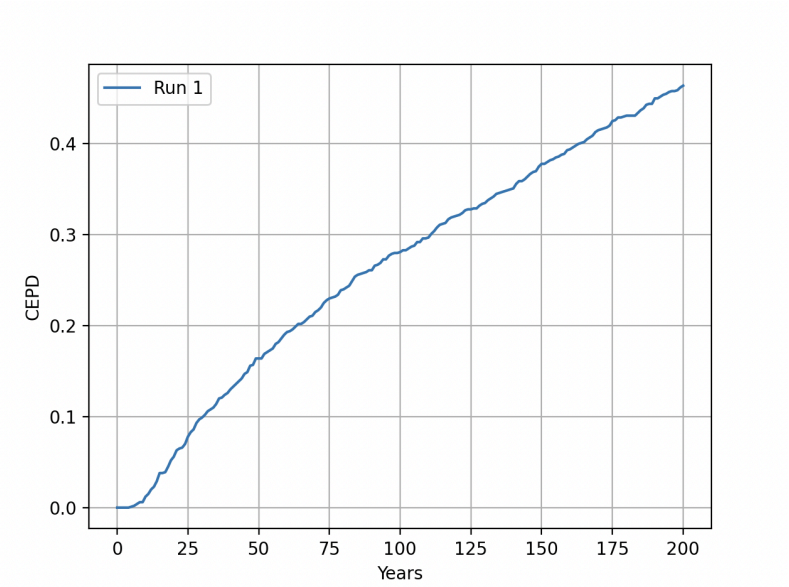
Results



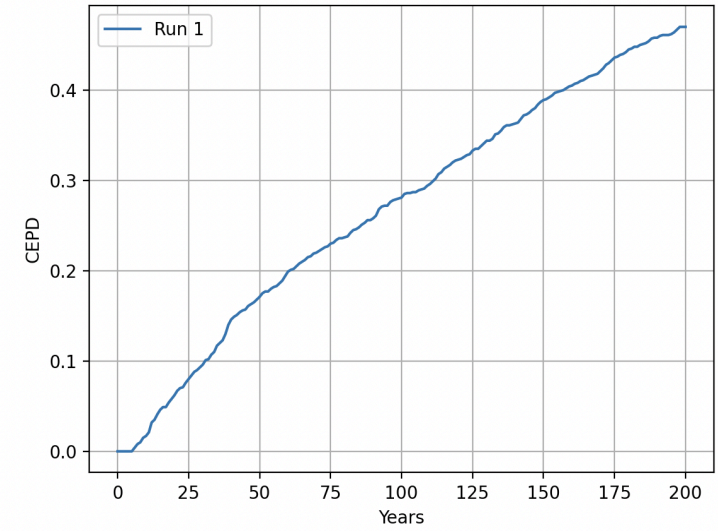
Above: Required plot 1: python3 penguin.py 1000 5
This plot is showing the three plots of the CEPD for three runs of 1000 simulations for 201 years with the given default parameters. This result makes sense because all of the outputs are pretty similar and follow the same pattern/curve.



(plot2)python3 penguin.py 1000 3 - 3 year El Nino cycle
This plot is for one run of 1000 simulations for 201 years with a 3 year El Nino cycle.



(plot2)python3 penguin.py 1000 5 - 5 year El Nino cycle
This plot is for one run of 1000 simulations for 201 years with a 5 year El Nino cycle.



(plot2)python3 penguin.py 1000 7 - 7 year El Nino cycle
This plot is for one run of 1000 simulations for 201 years with a 7 year El Nino cycle.

Required plot 2 *above (3 plots above)*: All the curves are pretty similar but a difference is that as the years go on, the larger El Nino cycles start to have slightly larger CEPD values, with the 7 year cycle having the largest. Otherwise, the shape of the graphs is pretty similar in all three. They are mostly linear with a slight rounded curve making them concave down.

Reflection
This program could be used to model similar circumstances in nature in the real world. Although the actual program used is probably much more complex, scientists use something similar to

model extinction rates and see different affecting factors and use this information to take action and make changes to preservation efforts. There are other nature related uses too, such as maybe weather or migration patterns.

Follow-up Questions

1. Both code snippets give the same result, but example 2 is definitely simpler because it doesn't process list 'a' many times in different formats. The first example can get confusing because it explicitly uses each index while the second example directly goes through the list.
2. We test code incrementally because it helps when debugging or figuring out issues or problems in the code. If you know which part is already correct, then you can cut down time spent on trying to find out what's wrong and instead directly go to the problem. Testing the code once after writing it all out causes confusion.
3. We use random numbers because it creates more variation and makes it more similar to the real world, because it is more random and unpredictable.
4. My favorite wild animal is a cheetah because they are very fast and are super cool animals.

Extensions

I did an extension with the purpose of supplementing the code with a plot by using matplotlib. First, I imported matplotlib.pyplot and mostly used what we learned from the lab. It got a little more complicated as I tried to plot multiple plots and I wanted it to use user input to know how many plots to make. The base code was already done such as getting the actual data, but I didn't realize I had to reformat it, so that is when I made a new function called plotCEPD to be called by the main function.

Acknowledgements

When I went to office hours, I got help from Meredith, Yubin, and Paisley. In my code, I also imported the libraries: random, sys, and matplotlib.pyplot. I also used online help and tutorials from w3 schools, the matplotlib website, the PYnative website, and stack overflow.