**Project #1**:

Switching Element Problem

**Mankirat Gulati**

ID: 111161128

CSE 346

**Source Code**

This program was written in **Python 3** and uses the libraries, **Numpy** to calculate the Bernoulli trials and **matplotlib** to plot the graphs.

*constants.py*

*A picture containing text

Description automatically generated*

File containing important constants for the program to work.

*simulation.py*

*A screenshot of a cell phone

Description automatically generated*

The full source code that handles the simulation and plotting of graphs.

**Program Algorithm**

* For each value of in or , I create two arrays, *passed* and *dropped* which contain the number of packets that went through and the number of packets that were dropped, respectively.
* I loop from to the () and perform () Bernoulli trials for each simulation.
* I get the number of packets sent by summing up the 1s in the *result* array.
* The *passed* array is updated with the number of packets sent or () if .
* The *dropped* array is updated with the number of packets minus the number of passed packets.
* The *results* dictionary is updated the probability as the key and the two arrays, *passed* and *dropped* as values.

**Graphs**

A screenshot of a map

Description automatically generated

**Flow Chart**

A screenshot of a cell phone

Description automatically generated

**Conclusion**

* Looking at first graph (Busy Outputs vs. Probability), we can see it first started out as an increasing linear relationship until around . From that probability onwards, the graph levels out and any probability above that points to an average of 3 busy outputs. With the second graph, (Packets Dropped vs. Probability), we can see a fairly consistent increasing linear relationship. This makes sense because the switch can only accept 3 packets at a time. If we increase the probability, we increase the number of packets that arrive. With more arriving packets, we have more dropped packets since the switch can only accept 3 at any time. Performance wise, this switch is fairly inefficient considering it can only handle a maximum of 3 packets at a time.