Zahir Choudhry
2 / 27 / 2022
HW#2 Write Up

1. (4pts) Provide detailed steps of how stack3.txt is evaluated using stacks.
2. (4pts) Will you be able to solve this problem using one or more `Queues`? Briefly explain why or why not you would use Queues over Stacks to solve this.
3. (2pts) What was the most proud thing you did in this homework? What was one thing that you wish you could do better?
4. If you received any external help (e.g. stack overflow links, blogs, worked with classmates, etc.), please list them here.

<br>

1. First, the stack will push 3 and 2.

   Then, it will read the * and multiply them to get 6, which will be pushed back on the stack.

   Then, 4 will be pushed on the stack and when the program reads the "^", it will return 6^4 (1296).

   Then it will push 1296 and 2 back onto the stack. After that, it will read the "-" and return 1296 - 2 = 1294, which will be pushed onto the stack.

   After this, 8 will be pushed on the stack, and when "/" is read, it will return 1294/8 (161.75).

   Lastly, the stack will push 161 and 2 onto the stack, and when "%" is read, will return 161%2, (1.75)

<br>

2. I would not use Queues over stacks to solve this particular problem. Due to the nature of Postfix (Reverse Polish) Notation and the way our input files are formatted, it would become considerably more difficult to read, process and compute the data. What makes it considerably more difficult to do with a queue is that you could only compute the first 2 numbers given, despite the order of operations. For example, say the data given is: 2 3 4 + /. You create 2 queues, one that initially holds all the data and another to hold all the numbers. When adding to the number queue, the order will be 3 -> 2 -> 4. So when it comes time to add, 3 and 2 will be added, when the intended result was 2 and 4.

<br>

3. I am proud of the speed I had with this project. As of writing this, the project is not due for another week, which leaves me plenty of time to study for midterms. I am also proud of my ability to logically map out how to perform the necessary calculations and write the proper code to do that. One thing I could have done better is initially create detailed and

coherent comments to explain my process. I had to go back and re-write most of my comments

4. Sites / Videos used:

Computerphile: Reverse Polish Notation:

https://www.youtube.com/watch?v=7ha78yWRDIE

Another RPN Video: https://www.youtube.com/watch?v=qN8LPIcY6K4