The most difficult part of this assignment was creating the flow of transitions and creating that relationship between the stages. I was not sure how to create that relationship and I could find any good examples on how to write it up in a program. I used the technique of creating a stage class just like I did a node class in my adjacency list to create the relationships between the stages. This strategy was very effective for me and it made the later processes of transitioning through stages very smooth. The overall implementation could still use some modification to make it easier on the user to run, and to allow for more versatility such as an input of chars rather than 1/0.

The reason I feel it is easier to implement a DFA that can process binary strings with three consecutive 1s is because you can create the relationship of the stages to fit that criteria. I can say the opposite for implementing 1^n0^n because the order of 0s 1s can be different which can cause it to be an unsuccessful input because the machine would need to track an unlimited number of possibilities.