Planning Document – Nicolas Sanchez Assignment 0: Found in the Shuffle

Planning Phase

Because each deck of cards is divided into 2 lines with 26 cards each, I will store each line into 2 separate arrays in order to prevent any input error. These 2 arrays will be combined later when needed in order to make everything easier.

Functions needed: One function, charInt, is needed to convert the characters in an array into integers and return a value based on the character. One function, findAscend, to find the greatest ascending sequence and another, findSuit, to find the greatest same suit sequence. charInt will take in 1 character and run it through a switch statement, then return the value. findAscend will combine the 2 char array deck halves into 1 integer array and sort through it finding the greatest ascending sequence. findSuit will combine the 2 char arrays into 1 string and sort through it finding the greatest same suit sequence.

Assistance Received

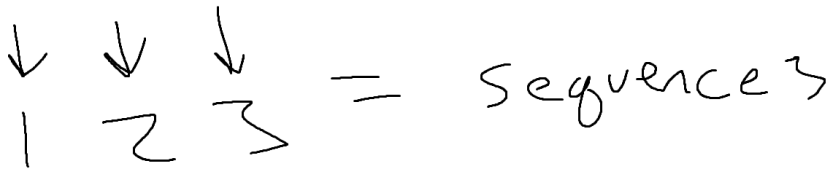
I received some assistance from the TAs in the UCF CS discord and during TA office hours

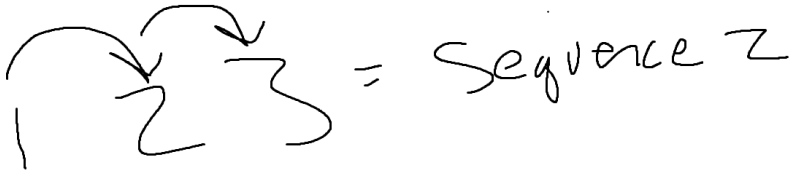
Debugging Phase

I now know to start this document when starting my program so I can record my debugging process as I go along. The main issue in my program was my misunderstanding of the ascending order requirement. I originally believed ascending order for this assignment would be any increase in value, but instead, it was an increase by 1. This caused some problems but after discussing it with the TAs in the discord I was able to switch everything to how it will be expected.

Testing Phase

Another issue I ran into was with the counter variable and how we were expected to use it. It's difficult for me to explain this so I will draw it. The drawing shows 2 ways of recording the counter variable and keeping track of the sequence. The top one would record 3 for 3 in a row and the bottom one would record 2 based on how I initialized the counter. I eventually realized the top one was the correct method when using different test cases and noticing a pattern of incorrect answers. Because I spent a lot longer on this assignment than I would've wished for, I wasn't able to create a program that would create test cases for me to use. Instead, I just used the quick and easy way of separating each card in a notepad file and randomizing the order myself. I "randomized" them in such a way where it would be easy for me to find sequences with my eyes so I could prove answers easily. Doing this I made 2 test cases that work and are formatted properly. Then, to prove the correct answers, I went through each card looking for the same suit and ascending sequences myself. I did this a couple of times and had other people do the same in order to reinforce my answers.





When I finished ironing out the rough edges of my program I went through and traced it all to reinforce my understanding of my own code. This was very helpful and I will start doing this earlier in the creation process so that I can reference previous code and fully understand/explain it. Because anything can happen, I retested my code after cleaning it up and tracing with all 4 test cases and had no problems.