Eduardo Nava

Project 1b documentation

The premise of this project was to create a program that will perform a specific set of functions. The functions that were required for this program are as follows: create program that will generate a random configuration of symbols, be able to read in an existing configuration into the program, print the configuration to the screen, print the configuration to a file, and have the user pick a specific location for a symbol and print it out along with its bonus value.

For the generation of the random configuration file, I decided to incorporate a random number generator as well as a few for loops. I decided to write a function that would use char arrays in order to accomplish this. The symbols file that was included was read into an array called symVal. This would then be passed into the function, and a randomly generated number ranging from zero until the last element in the array (6 elements in this case) , would pick a symbol to be outputted to a default configuration file. The function would then read in this file into another 3-d array “ace”, from which the configuration for that instance of the program is created.

Printing the configuration was just a matter of using cout within a for loop to output the elements in order to the terminal. Outputting a configuration to a file involved creating another char array. The user would then be prompted for a file name, and this file name would be imputed into the new char as a string. This string would then be used to open a new file with that name, and the “ace” array holding the current configuration would be outputted to it. Reading in a configuration from a file was similar to outputting to one. Extraction operators were used once the file was open and the symbols were transferred to the “ace” array, making it the current configuration.

The most difficult part of the project was figuring out the need to write a sting comparison function in order to complete the last objective. The last requirement involved asking the user for a reel and stop number which would be stored in a couple int variables. These variables were then passed to a function that would compare the string contained in the ace[stop][reel] with every string in the “symVal” array. When a match is found the program outputs the symbol and the corresponding bonus value.

If I were to make any changes to this program I would probably want to find a more elegant solution to reading in text from a file line by line. The current way I read in a file line by line involves knowing the structure of the text within the file that is being read. If possible I would like to implement a more automatic approach in the future.