Alex Maass

CS 2024- C++ Programming

Assignment #5

Dietel & Deitel Exercise 10.10 & 10.11

For this assignment we were asked to create a deck of virtual playing cards. The deck of playing cards would be made of individual cards that would have a unique face and suit value. The deck would then shuffle these cards to produce a normal random deck of cards and the main program would deal 5 cards to the user. The main program was then to further check these 5 dealt cards to see if they met certain criteria, such as if any pairs were present or if the hand was a straight.

The way I solved this problem was by creating Card and DeckOfCards classes. The Card class would have two fields, face and suit, which would represent the cards suit and face values. The Card would also have a toString function to print out both values of the card. There are getter functions for both fields. The DeckofCards class is mainly composed of a vector of cards. The main program will generate all 52 cards in a deck sequentially and place them into the DeckOfCards. This is done by having two static arrays associated with class Card that contain all the possible values for face and suit. The cards are put into the deck vector by the function push\_back(). Once all 52 cards are placed in the deck, the deck will then shuffle itself. The deck shuffles itself by iterating through the vector and for each card in the vector the shuffling algorithm swaps the card in the original location and the card in the random location in the deck. By doing this, the deck gets completely shuffled with cards being moved around many times or possibly not at all, as a normal deck would get shuffled. Then 5 cards are dealt to the user using the dealCard function which looks at the next card in the deck. These 5 cards are printed out and now the hand detection system comes into play.

To detect various hands, the main idea is to iterate through the hand and compare that card to all other cards left in the hand. Since order doesn’t matter and the cards are shuffled, the iterator only needs to go 1 way through the hand to find all possible matches. Nested for loops are used to look at multiple arrangement of cards. To detect a flush, the system looks at the suit values of all 5 cards and asserts that there is a flush if all 5 suit values are the same. The straight detection is a little bit more complicated. Since the face values for all the cards are encoded in string form, they must first be changed to numbers. This is done by comparing the face values to all the possible face values and since the face value array is ordered, the number of the card can be matched to the index of the array where the match occurred plus 1. Once all the string face values have been converted to numbers, the numbers are thrown into another vector that is then sorted using the sort function provided by the algorithm file in the C++ library. Once the vector is sorted, it is easier to check that all the numbers are in consecutive orders and if they are, the system asserts that a straight has occurred.

This assignment taught me a lot. It taught me the basics of vector and array manipulation along with a lot of string manipulation. Completing this assignment has given me a lot of experience in working with strings and allowing me to use vectors to accomplish some very complicated task in a simple way. It also allowed me to explore the various C++ libraries looking for built in functions so that I would not have to redefine (overload) them.