Question 7

I have split up the program into 3 separate files due to the 3 different containers. I chose to generate and print 100,000 characters using arrays, linked list, and vectors. My hypothesis was that the array would be the fastest because I felt that it was the simplest data structure and it didn’t have to do many operations. My hypothesis was in fact correct. The array had performed the operation in the fastest speed. In the array program, I create a function called genCharArray that takes in a size as a parameter. The function is very simple as it creates a new character array inside of he function, fills it with random characters with a for loop and outputs it all in one function. The linked list program features all of the code that is required for a linked list. This is the same code that we used in problem one(The linked list is explained thoroughly in the white paper for problem 1 and therefore I did not repeat it here). In the main function of this program I create a empty linked list and set the head and last pointers to be null. I create a for loop and insert a random character in the linked list until the for loop is complete. The linked list did indeed take the longest time to complete as it has more operations to complete. For the vector program, I used the same code as we did for problem one and therefore I am not going to repeat the explanation here. I create a function called generateVecWithChars that takes an integer size as a parameter. This function immediately creates an F type vector of the size that was passed through. Then there is a for loop in which the vector is adding a random character at each index outputs the character until the for loop is over. These programs all generated 100,000 characters and outputted them in which the array topped off with the fastest speed.