Brian Laccone
CS 362
5/6/2018

Random Testing Quiz

The first thing that I did was try and figure out what exactly the testme.c file was trying to do. It took me a few minutes but I eventually figured out how the test was being run. Creating the inputChar() function was pretty simple as it simply needed to hit one character to increase the state. The first thing I had to do was find the correct range of ASCII characters. I then set a variable equal to a random char in that range and returned it. My test would consistently hit state = 9 quickly. I have to try really hard to see it display any state before that point as the program runs incredibly fast.

```
char inputChar()
{
  // TODO: rewrite this function
  int minNum = 32;
  int maxNum = 126;
  int range = maxNum - minNum;
  char randChar = (rand() % (range + 1)) + minNum;
  return randChar;
}
```

Once the state is equal to 9, the program then continues until the string = “reset”. The inputString() function does the same thing as inputChar() but instead of only creating one random character, it creates 5 random characters and stores them in a string and then returns that string. I started out with the randomized characters being in the range of the whole alphabet and it found the error before 5 minutes, most of the time, but occasionally it would take longer than 5 minutes, so I decided to make the range cover the target statement. The test finds the error much faster now.

```
char *inputString()
{
  // TODO: rewrite this function
  int minNum = 101;
  int maxNum = 116;
  int range = maxNum - minNum;
  char string[6];
  int i;
  for(i = 0; i < 5; i++){
```

```
    string[i] = (rand() % (range + 1)) + minNum;
  }
  string[5] = '\0';
  char *randStr = string;
  return randStr;

}
```