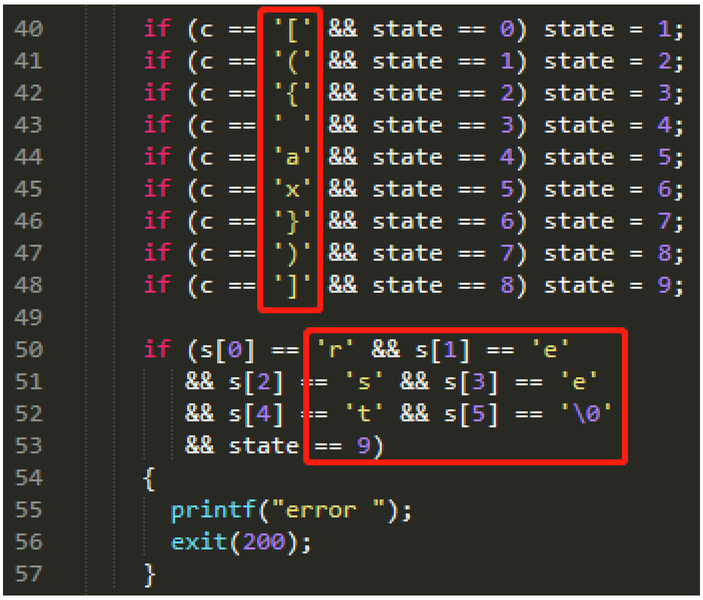
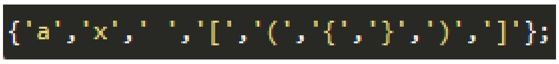
CS 362 Random Testing Quiz Write Up
Yang Yang
10/25

After observing the if statements inside the while loop:

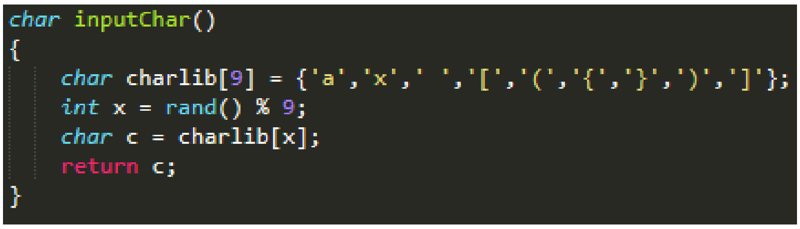


```
    if (c == '[' && state == 0) state = 1;
    if (c == '(' && state == 1) state = 2;
    if (c == '{' && state == 2) state = 3;
    if (c == ' ' && state == 3) state = 4;
    if (c == 'a' && state == 4) state = 5;
    if (c == 'x' && state == 5) state = 6;
    if (c == '}' && state == 6) state = 7;
    if (c == ')' && state == 7) state = 8;
    if (c == ']' && state == 8) state = 9;

    if (s[0] == 'r' && s[1] == 'e'
       && s[2] == 's' && s[3] == 'e'
       && s[4] == 't' && s[5] == '\0'
       && state == 9)
    {
      printf("error ");
      exit(200);
    }
```

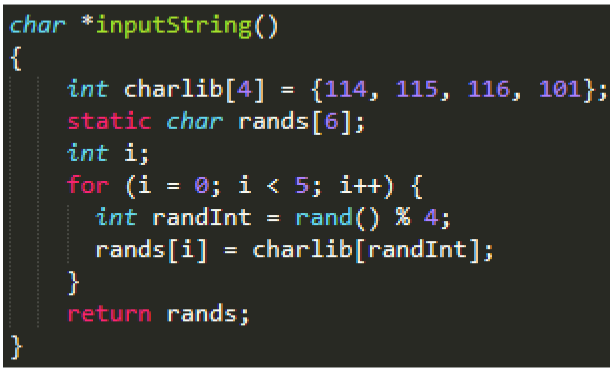
I notice that the characters we care are:  `{'a','x',' ','[','(','{','}',')',']'};` and the characters in “reset”. Randomly select one character from the list and eventually, the state will be increased to 9. Then randomly generate a combination of strings using {r, e, s, t}. Ultimately, if the random string is “reset”, the coverage is complete.

So for inputChar(), I just randomly return a char from the list of characters we care;



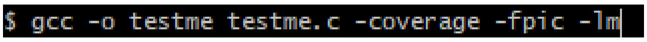
```
char inputChar()
{
    char charlib[9] = {'a','x',' ','[','(','{','}',')',']'};
    int x = rand() % 9;
    char c = charlib[x];
    return c;
}
```

Similar method for inputString:

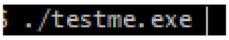


```
char *inputString()
{
    int charlib[4] = {114, 115, 116, 101};
    static char rands[6];
    int i;
    for (i = 0; i < 5; i++) {
      int randInt = rand() % 4;
      rands[i] = charlib[randInt];
    }
    return rands;
}
```

Then I ran following scripts:

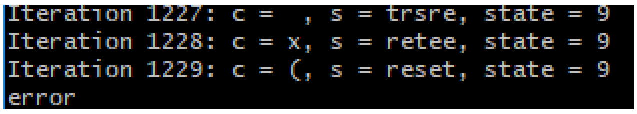


```
$ gcc -o testme testme.c -coverage -fpic -lm
```



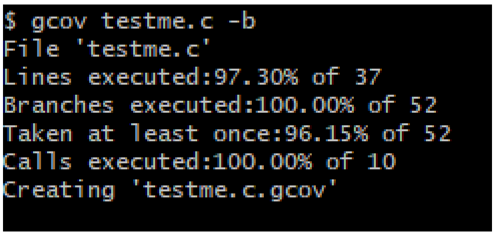
```
$ ./testme.exe
```

Error occurs:



```
Iteration 1227: c =  , s = trsre, state = 9
Iteration 1228: c = x, s = retee, state = 9
Iteration 1229: c = (, s = reset, state = 9
error
```

Then use gcov:



```
$ gcov testme.c -b
File 'testme.c'
Lines executed:97.30% of 37
Branches executed:100.00% of 52
Taken at least once:96.15% of 52
Calls executed:100.00% of 10
Creating 'testme.c.gcov'
```