```c
/*
 * Name: Albert Diaz
 * Course: CS 362 W2018
 * Title: Random Testing Quiz
 * Date: 2/4/2018
 */

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<time.h>


/*
    The inputChar() will returns a random integer corresponding to ASCII table.
    It utilizes the rand() and has a range from 32 to 127.  It is able to
    quickly provide a random number and with the small range of numbers finds
    state 9 quickly as well.
 */
char inputChar()
{

    return (rand() % 95) + 32;
}

/*
    The inputString() will build a random string up of length 6 characters.
    The first 5 characters will be filled with a random integer and the 6th
    character will be a '\0'.  It took billions iterations for my test to find
    the error.  This shows how much increase in testing coverage is required
    when a few more values are required to be checked.  Reducing the range of
    characters to 97 to 123 used to provide a closer chance to target was
    necessary to make the testing complete faster.  It took only 2.15 million
    iterations to find the error versus what could have taken hours to days
    with the range of 32 to 123.
 */
char *inputString()
{
    int i;
    char rest[5] = "rest";

    static char string[6];

    for(i = 0; i < 5; i++)
    {
        string[i] = (rand() % (123-97)) + 97;
    }
    string[5] = '\0';
    return string;
}

/*
    The testme() utilizes both the inputChar() and inputString() developed with
    a series of 'IF' statements to obtain state 9 and check if a string value
    is equal to 'reset' and has a state value of 9.  Once those two conditions
    are met the function will print "error" and exit.
 */
void testme()
{
    int tcCount = 0;
    char *s;
    char c;
    int state = 0;
    while (1)
    {
        tcCount++;
        c = inputChar();
        s = inputString();
        printf("Iteration %d: c = %c, s = %s, state = %d\n", tcCount, c, s, state);
```

```
70
71          if (c == '[' && state == 0) state = 1;
72          if (c == '(' && state == 1) state = 2;
73          if (c == '{' && state == 2) state = 3;
74          if (c == ' '&& state == 3) state = 4;
75          if (c == 'a' && state == 4) state = 5;
76          if (c == 'x' && state == 5) state = 6;
77          if (c == '}' && state == 6) state = 7;
78          if (c == ')' && state == 7) state = 8;
79          if (c == ']' && state == 8) state = 9;
80          if (s[0] == 'r' && s[1] == 'e'
81               && s[2] == 's' && s[3] == 'e'
82               && s[4] == 't' && s[5] == '\0'
83               && state == 9)
84          {
85              printf("error ");
86              exit(200);
87          }
88      }
89  }
90
91
92  int main(int argc, char *argv[])
93  {
94      srand(time(NULL));
95      testme();
96      return 0;
97  }
```