932526065
Yi-Chiao,Yang

CS562- TSTL Final report

In this project, I used single linked list as my SUT. There are total fifteen functions in this python library, and I tested each function in tstl file. There are three sections in tstl file, including utility functions, pools and actions. In order to satisfy the syntax in tstl file, I design some utility functions for testing. It is hard for me to design some utility functions in tstl file because I have to test whether this correct or not in that linked list python library. I could not directly design utility functions in tstl file, so it takes me lots of time to design utility functions on python library file and move to tstl file. Sometimes, I need to modify these utility functions which I write in python library file because some parameters in python and in tstl are different.

In the data structure of single linked list, there are three basic functions, including generation, insertion, and deletion. For the generation, it means that it can create an empty list to do following actions. For the insertion, it means that it can add data in to this list. In this python library, the author also designs some functions about insertion. For example:

`append(self, data)`**:** Add a data at the end of the list.
`updateIndex(self,index,data)`**:** Update a data in a index position that we want to change it
`insertBeforeIndex(self,index,data)`**:** Add a data before a index position that we want to change the list before this index.
`insertAfterIndex(self,index,data)`**:** Add a data after a index position that we want to change the list after this index.
`insertAfterEveryData(self,data,dataToInsert)`**:** Add a data after every data
`insertBeforeEveryData(self,data,dataToInsert)`**:** Add a fata before every data

Some functions about deletion:
`deleteIndex(self, index)`**:** Delete a index position
`deleteData(self, data)`**:** Delete a data that exists in this list
`deletePtr(self,ptr)`**:** Delete the pointer that we move to next index
`deleteAllData(self,data)`**:** Delete all data
`deleteList(self)`**:** Delete a list that you mention to delete it.

Bugs:

I admire the author because I have not found any bug in this python library. The author's thought is cautious. In each function, the author set the code that returns false when users input wrong values. If I have to seriously discuss this python library, I would like to analyze the returnIndex function. This function can search a data what we look for its index. If this function successes, it returns an index position where the system finds it first. However, if the following index position contains the same data as the previous index position, this system cannot return all of index positions that contain the data we look for. In fact, it can be a bug or cannot be a bug in the python library because it actually returns a correct result. Thus, I did not report it as a bug in this section.

Tester's work:

There are three main works I am done for writing a tstl file. The first thing is to understand the python library that I choose it as SUT and processes in TSTL. The second thing is to design logic syntax sentences for testing. Sometimes, the tstl file can compile successfully without any error. It looks like all are correct, but the number of percent of coverage do not increases, or it decreases. In this situation, it has a mistake in syntax. It is hard to correct it, and I spend lots of time modifying it. The final thing is to analyze what I am done for this python library. If I am the writer of this python library, how to improve this python library? The purpose of TSTL is to assist programming designers to find bugs. That is what we done for this project.

Wrong with tstl:

A weird thing I found that is in the initialization. The initialization means that construct for an empty linked List and takes a list and produces a linked List containing all elements in the list. If the TSTL files only initializes the class, and do nothing in the following task, it shows the error:

```
COV: CLEARING OLD DATA
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
Traceback (most recent call last):
  File "randomtester.py", line 323, in <module>
    p = R.randint(0,len(acts)-1)
  File
"/System/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7
/random.py", line 242, in randint
    return self.randrange(a, b+1)
  File
"/System/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7
/random.py", line 218, in randrange
    raise ValueError, "empty range for randrange() (%d,%d, %d)" %
(istart, istop, width)
ValueError: empty range for randrange() (0,0, 0)
```

If we write a basic linked List API call after initialize the linked List, it won't show any bug. I am not sure what the reason causes it. However, if the tstl code is correct, but it dos not have an API call, it should still can compile the TSTL file. Also, I test a little bit the initialization in the python file to figure out what happen if I only write the initialization in the python code, it still can work, but does not show any result or error. Thus, I think that we could add a warning if testers only test the initialization.

Another weird thing happened when I design my tstl file. As we know that we have to define the value pools that are used in testing. A test in tstl is a sequence of assignments and calls, and the value pool contains the values used. The weird thing is about the name that is defined by myself. Generally, it is named as capital letters. In my tstl file, it also can work when it is named a lowercase letters. In my experience in C language, because it is easy to distinguish values and parameters, I am used to use " - " when I give value or parameter a name. If I name pools' values as <INT_> or <1_INT>, it shows this error when I compile randomtester file.

```
   Traceback (most recent call last):
  File "randomtester.py", line 1, in <module>
    import sut as SUT
  File "/Users/Chiaoysbaby/tstl/generators/sut.py", line 3066
    self.p_<aa_aa> = {}
                   ^
SyntaxError: invalid syntax
```

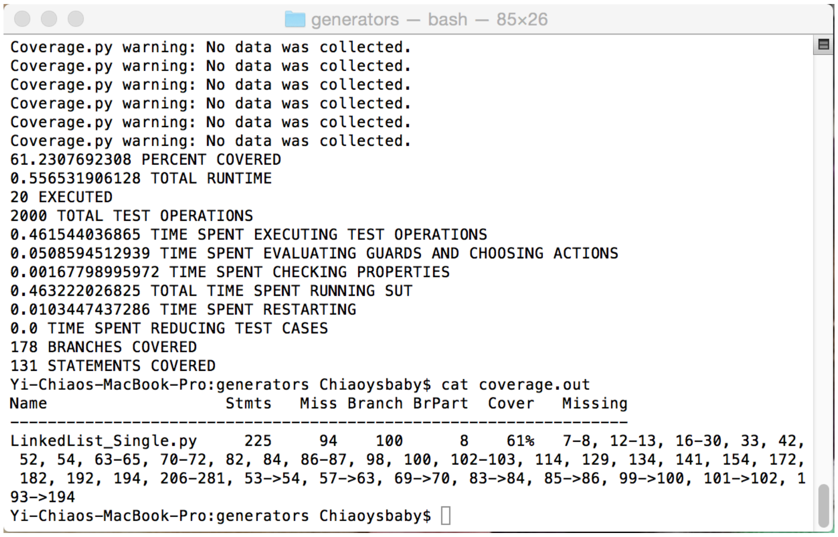
It shows that I have a syntax error, but I can use dash to name values or parameters whatever in python or other language. I am not sure what the problem causes this syntax error. The reason might relate to the randomtester file.

Good with tstl:

TSTL is a good tool to assist programming designers to find their bugs. It provides abundant of random test for programming designers. In my section of tester's work, I point out that three things tester need to do. Only three processes in this testing work, it provides people a clear method to test their code. In conclusion, TSTL is simple, clear, functional test tool.

Coverage summaries:

This picture is from my coverage.out file. It finally cove percent 61 of the code, and it has total 2000 test operations.



```
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
61.2307692308 PERCENT COVERED
0.556531906128 TOTAL RUNTIME
20 EXECUTED
2000 TOTAL TEST OPERATIONS
0.461544036865 TIME SPENT EXECUTING TEST OPERATIONS
0.0508594512939 TIME SPENT EVALUATING GUARDS AND CHOOSING ACTIONS
0.00167798995972 TIME SPENT CHECKING PROPERTIES
0.463222026825 TOTAL TIME SPENT RUNNING SUT
0.0103447437286 TIME SPENT RESTARTING
0.0 TIME SPENT REDUCING TEST CASES
178 BRANCHES COVERED
131 STATEMENTS COVERED
Yi-Chiaos-MacBook-Pro:generators Chiaoysbaby$ cat coverage.out
Name                       Stmts   Miss Branch BrPart  Cover   Missing
-----------------------------------------------------------------------
LinkedList_Single.py     225     94    100      8    61%   7-8, 12-13, 16-30, 33, 42,
 52, 54, 63-65, 70-72, 82, 84, 86-87, 98, 100, 102-103, 114, 129, 134, 141, 154, 172,
 182, 192, 194, 206-281, 53->54, 57->63, 69->70, 83->84, 85->86, 99->100, 101->102, 1
93->194
Yi-Chiaos-MacBook-Pro:generators Chiaoysbaby$ 
```