

Bugs Report

In this project, I find a bug about my system under test python file. In the first function `create_by_head (self, datas)`, I write a `tstl` function to test whether it can successfully insert a list into the Linklist.

“Node(data[0])” Bug

First, I defined the `length` function in the `tstl` file because my test python file has no function to get the size or length of the Linklist. So, I defined a function called `length(self)` to get the length of the Linklist.

Test Function:

```
~<LINKED>.create_by_head(<LIST>) => (length(<LINKED,1>) ==  
PRE<(len(<LIST>))>)
```

In the test function, `LINKED` is the target Linklist, `LIST` is the list we want to insert. First we initial a empty list and randomly append `<INT>` in the list. Then length of `LINKED` should be equal to the length of `LIST` we get because the `LINKED` is empty at the beginning. So, if two length equal, it means we insert successfully.

When I test this function, I initialize the empty `<LIST>:[]`, then, appending `<INT>` in the `LIST`. When executing `create_by_head` function, it would report bug. I called it “Node(data[0])” bug.

```

generators — bash — 80x24
FINAL VERSION OF TEST, WITH LOGGED REPLAY:
LINKED2=linklist.LinkList()    # STEP 0
LIST0=[]                        # STEP 1
LINKED2.create_by_head(LIST0)   # STEP 2
ERROR: (<type 'exceptions.IndexError'>, IndexError('list index out of range',),
<traceback object at 0x1007d0ab8>)
TRACEBACK:
  File "/Users/shikai/tstl/generators/sut.py", line 3963, in safely
    act[2]()
  File "/Users/shikai/tstl/generators/sut.py", line 2710, in act116
    self.p_LINKED[2].create_by_head(self.p_LIST[0])
  File "/Users/shikai/tstl/generators/linklist.py", line 53, in create_by_head
    self.head = Node(datas[0])
STOPPING TESTING DUE TO FAILED TEST
0.0362520217896 TOTAL RUNTIME
1 EXECUTED
45 TOTAL TEST OPERATIONS
0.0067195892334 TIME SPENT EXECUTING TEST OPERATIONS
0.00154900550842 TIME SPENT EVALUATING GUARDS AND CHOOSING ACTIONS
3.5285949707e-05 TIME SPENT CHECKING PROPERTIES
0.00675487518311 TOTAL TIME SPENT RUNNING SUT
0.00119304656982 TIME SPENT RESTARTING
0.024512052536 TIME SPENT REDUCING TEST CASES
10-251-120-230:generators shikai$

```

It says when we insert an empty list in the Linklist, it would crashed due to the Node(datas[0]). So, I write a print function in python file to print the result of inserting empty list. It also crashed and reported bugs like this:

```

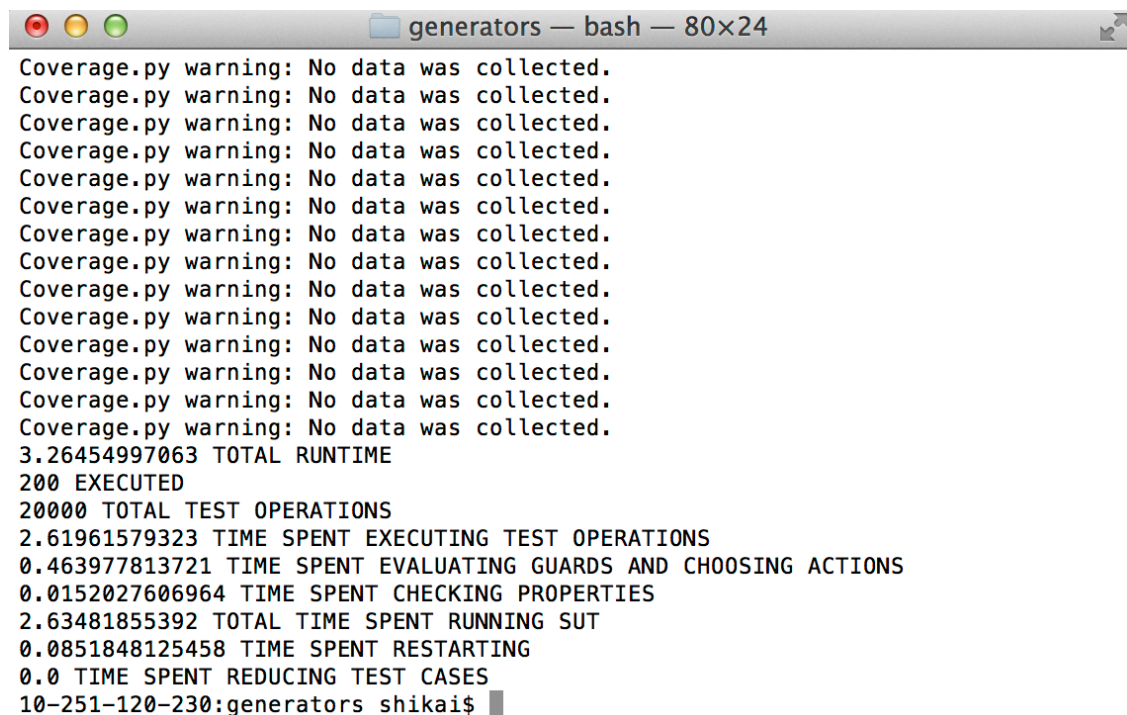
generators — bash — 80x24
10-251-120-230:generators shikai$ python linklist.py
Prepared inserted list:[]

Traceback (most recent call last):
  File "linklist.py", line 317, in <module>
    llst_i_1.create_by_head(intdatas_1)
  File "linklist.py", line 53, in create_by_head
    self.head = Node(datas[0])
IndexError: list index out of range
10-251-120-230:generators shikai$

```

Therefore, it is the bug of this python file. Then, I analysis the reason it report this bug. I see in the `create_by_head(self, datas)` function, the author first define `self.head = Node(data[0])`, but he does not consider when `datas` is empty. When `datas` is empty, `data[0]` would be empty too, and `head` would also be empty. This function would failed. That's the reason why it cannot insert empty list to the Linklist.

So, I changed `<LIST>: = []` to `<LIST>: = [1,2]`. It means I initialize the list as `[1,2]` rather than empty. It can insert list successfully.

A terminal window titled "generators — bash — 80x24" showing the output of a test run. The output consists of 14 lines of "Coverage.py warning: No data was collected.", followed by a summary of test statistics: "3.26454997063 TOTAL RUNTIME", "200 EXECUTED", "20000 TOTAL TEST OPERATIONS", "2.61961579323 TIME SPENT EXECUTING TEST OPERATIONS", "0.463977813721 TIME SPENT EVALUATING GUARDS AND CHOOSING ACTIONS", "0.0152027606964 TIME SPENT CHECKING PROPERTIES", "2.63481855392 TOTAL TIME SPENT RUNNING SUT", "0.0851848125458 TIME SPENT RESTARTING", "0.0 TIME SPENT REDUCING TEST CASES", and finally the prompt "10-251-120-230:generators shikai\$".

```
generators — bash — 80x24
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.
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.
Coverage.py warning: No data was collected.
Coverage.py warning: No data was collected.
3.26454997063 TOTAL RUNTIME
200 EXECUTED
20000 TOTAL TEST OPERATIONS
2.61961579323 TIME SPENT EXECUTING TEST OPERATIONS
0.463977813721 TIME SPENT EVALUATING GUARDS AND CHOOSING ACTIONS
0.0152027606964 TIME SPENT CHECKING PROPERTIES
2.63481855392 TOTAL TIME SPENT RUNNING SUT
0.0851848125458 TIME SPENT RESTARTING
0.0 TIME SPENT REDUCING TEST CASES
10-251-120-230:generators shikai$
```

That's the first bug I found in this project. Also, the same bug would appear in the `create_by_tail(self, datas)` function.

The second bug of the python file is the `clear()` function. It also caused by "Node(data[0])" bug. I thought after executing `clear()` function, the length of LINKED should be 0. So, I write this: `~<LINKED>.clear() => (length(<LINKED,1>) == 0)`. But it would report bug, which says `length(<LINKED,1>)` not equals 0. So, I thought in this file, when it is init, it defined `self.head = Node(0)`. So, its length is at least 1 at the

beginning. So, I should write `(length(<LINKED,1>) == 1)`. It can be executed. But in fact, when we clear a Linklist, the length should be 0. If this is the black box test, the function would be considered as failed. But in this project, I have read his codes, so I consider it successfully clear the whole Linklist.

Progress

Last time I have only finished testing `create_by_head` function. This time, I have finished testing `create_by_tail`, `clear()`, `is_empty()`, `getData_by_index()` functions.

For the `is_empty()`, `getData_by_index()` function I didn't find any bugs. To test `is_empty` function, I let `a = ~<LINKED>.is_empty()`, then check whether `a` is True or False.

For `getData_by_index()`, I executed `create_by_head` first, then change the random list to a fixed list. Then, I checked `getData_by_index(0)` whether it equals to the first data of the list. Also, I checked `index(1000)`, whether it equals to False.

Progress by end of term and coverage

By the end of term, I will finish the other six functions about the Linklist. I will define some other functions in `tstl` file to test these functions. In the end, I'm going to cover more than 50 percent codes. Also, I noticed a small bug about the `tstl` and I need to test more to make sure. And I will include it in the final report if it corrects.

Quality of the SUT

The quality of the SUT is between poor quality and well done. Because his initialize function is not good. He didn't consider when list need to be inserted is empty, what should codes do. However, it is not a big problem because this function can still insert non-empty list into the Linklist. And other functions like `getData_by_index`, he writes well. No more bugs found until now. "`Node(data[0])`" is the only bug I found now.