Student ID : #932262940 Name : Xuan He

Bug/progress report

I selected the btree library as the system under test. So far, I didn't find any bug about the library. I have tested three functions of the OOBTree model in the library. I generate several btrees and randomly generate some strings and numbers as keys and values. Using the keys and values as input to test the insert and update function, and using the keys as input to test the pop function, and then check whether operations are successful.

Compared with the first version of my TSTL file, I add some property to each test function in order to check whether the test function is correct. After each operation, I check whether the btree is still a btree (*check(<btree,1>) == None*). For insert function, I check whether the length of the btree is increased by 1 after insert operation (*len(<btree,1>) == pre<(len(<btree,1>))> + 1*) or the key that I want to insert already exists in the btree (*<btree,1>.has_key(<key,1>) > 0*). For update function, I check whether the length of the btree is increased by 1 after update operation or the length of the btree does not change after update operation (*len(<btree,1>) == pre<(len(<btree,1>))>*), and I also define a new python function to check whether the data of the btree is updated after update operation (*check_update(<btree,1>,<key,1>,<value,1>) == 1*). For pop function, I check whether the length of the btree is decreased by 1 after pop operation or the btree does no contain the key that I want to delete (*<btree,1>.pop(<key,1>,0) == 0*), and I also check whether the btree contains the key that is the input of the pop function after the pop operation (*<btree,1>.has_key(<key,1>) == 0*). In addition, I also add some logs to my TSTL file which can show the process of testing.

So far, I have test three functions the OOBTree model in the library. Next, I will

consider whether I need to add more property to these functions. For update function, I just test the situation that there is only one pair (key, value) in the input argument, so I will test the situation that the input argument of update function is a list of pairs (key, value) in the future. What's more, I will also test another two function -- maxKey and minKey -- by end of term.

The quality of the SUT is very good. I think I did well tested for the insert, update, and pop function. I use different kinds of argument as the key and value of the btree nodes, such as string and int, and check all the changes of the btree after each operation, and I still did not find any bug for it. Therefore, it is very good code where finding bugs is likely to be hard. Moreover, it is also a widely used library, as the office website shows that it has 16521 downloads in the last month.

The Btree library is a very big library. There are five modules, and for each module there are four data structures. My test just focuses on the OOBTree module and BTree data structure. In this part, it also contains many functions. I select 5 functions to test, which coverage about 30 percent of the whole functions in this part.