

CS 362
Zixuan Feng
Final project

Test experience

During this term, in CS 362, we learned a lot about how to debug and test a program. Even we know that programming is a key step when we develop software, but test is indispensable. In this term we learned various ways to test software, like unit test, random test. Fixing bugs sometimes is easier than finding bugs, using these ways to test software will help us find bugs easily.

In the beginning of this term, we were going to test the dominion game; this program is a very large program. Even we know there are some bugs in there, but we still do not know how to use tools to help us. It would be a very hard to find bugs. As for a over 1000 lines program, we need to get familiar with the code. So our first assignment helped us learn and get familiar with the code. After assignment 1, I was getting familiar with the code, and I searched the rules of this game. Knowing the rules and get familiar with code is necessary for the testing. According to the Agans' Debugging Rules, we need to understand the system at first. So, I just follow the rule and deeply learned the code, play this game online. After few days, I know the rules of the game, and I read the code line by line. This helped me to write test for the program easily in the future.

Due to the large program, I used various ways to test the codes, including unit test, random test and the whole program test. In this term I also learned to know how to use the GCOV to check my test's coverage, this is very helpful. But this term's goal is going to learn use the various test in proper way. Also according to the Agans' rule, the heart of debugging is dividing and conquer, so learn how to use test tools is one of the important issue.

In this assignment 2, we learned how to use card test and unit test, I learned when I want to test some small function, I need to use the unit test, this is very helpful especially when I want to test some specific card effect, we could just set up a small range and use the unit test to test it. During the assignment 2, I test four units; I found a bug in my dominion, which I created in the assignment 1, in the smithy function. In the line of 1447 in dominion, I changed the card number. Through my card test, I found this bug; and my bug test told me the card number was wrong. This was inspiring me, and pushed me moving forward to learn how to debug in the future. In the unit test and card test, according to the lecture, the unit test is a very first phase, and it need to done by developers of modules. I think the unit test is one of the first tool for every programmer, and this test strategy will help programmer find a bug in a small range by using a easier way.

What is more, in the assignment 3, I learned how to use the random test, I really like

the random test, and this is a great idea for testing, because random test is good tool for test large program. I used the random test to test three different functions, adventurer, village and the smithy, I planed to let my coverage up to 100%, compared with the unit test, random test is easy to 100%, I set up to 100 times to test. And I just realized that it is good for complex and large data program. Especially for some loops in function, if the coverage is high, random test is helpful. When I tried to test my adventurer, I know this function is used for whole game, so I used a print out to read my test result. And I used the GCOV to see my coverage. Combined with these two results, it will easily to help test my result. After test the codes, I found two bugs I created in the assignment1. Firstly, I found the actions wrong in the village function, and there is another actions wrong in the adventurer function. After test, I learned how to use the random test, and I learned lot advantages of the random test. We could get the code coverage, and then we could adjust the code to generate inputs since the inputs are randomly. After test, we could just break the code in anywhere and inspect the bug. All these steps will help us improve the code correction. Some times when we use unit or card test, we need to several times to make sure the bugs, but random test will automatically repeat a lot times to help us find bugs.

At last, in the assignment 4, I learned how to test the whole game, in the assignment 4; we are focusing on test whole dominion game. In the assignment 4, I deeply know the differences between the partial test and whole game test. We could easily use unit and random test to test the partial function. As for the random test, it will change the data randomly, and this could give us a clear and complete result. As for a whole game test, I learned how to enlarge my test branch. When I was doing my assignment 3, I know how to use the random test, but in this time, I learned how to enlarge the random inputs for the entire game to make sure random inputs cover as much as possible. When I test the assignment 4, I know the coverage is the key point, higher coverage, we could get complete test. According to the lecture, there is a saying, “We should have an argument for any kind of coverage.”

Testing classmates' program

Jiangzhi' s Dominion test

Onid name: jiangzh

When I first time to see Jiang Zhi's dominion, I cannot tell any wrong with it, since the code is over 1000 lines, I cannot tell it or just compare with mine. It will be a complex way for me. So I just used my entire test program for his dominion. Firstly, I used my card test to test his code. I used my card test, test the smithy, it past. I cannot find any bug from it. And turn to use the second card test, village test. My test showed the print, village went wrong, the card actions wrong. And I just break my test program, and print out the result step by step, I found there is a bug in his dominion program line 1336, I found he let the number of actions become a wrong number. After I communicated with him, he proved he has a bug in village function. And then

I just keep going to other tests. I test used my cardtest3 to test his steward function. I did not find bugs. And then I test great hall by my cardtest4. I did not find any bugs. And I used unit test to test his other functions, I test his update coin, trash card, draw card and end game. His dominion program past all my unit tests. Thirdly, I used my random test to test his program, first I used the random test to test his adventurer, I find another bug from my test result. I test five times for his adventurer just want to make sure my test is correct. Every time I could find the bug in his adventurer. I checked my coverage, I find my random test for adventurer, the coverage is around 100%. So I just make sure bug exists. And I just follow the step of debugging, break my test and print our result step by step, and I found there is a bug in the line 1255, he let the Z equal to 1. But this z need to initialized first. And then I used my random test 2 to test smithy again, and my coverage is around 68, I did not find any bugs. And then I used my random test 1 test his village again, I found the bug again. Compared with my card test, I though my random test is effective. According to my test result, I found the number of actions have bugs. And then I used my test dominion to test the whole program, I checked my coverage, it is up to 70%,

Jiu Jiawei' s Dominion test

Onid name: liujiaw

Due to the large data in this program, I planed to still use all my test programs to test his dominion program. Firstly, I used my card test 1 to test his smithy function, and then I found a bug in his smithy function. Just want to make sure, I just test several times, and I make sure he has a bug in his smithy function. After debug process, I found there is a bug in his dominion line 662, he changed the cards of numbers. And then I used my card test 2, and I found another bug in his village function. After several times, I think my test program is correct, I think his program has bug in his village function. In the same way as I test Jiang Zhi's program, I found there is a bug in his dominion line 679, he also changed the number of actions. And then I used my card test3, to test his steward function, it passed. And then I test his great hall function, he just past my great hall test. And then I used my 4 unit tests, and his program past all my test. I used my random test to test his program again, His program past my adventurer test. But his program did not past my random test card 1 and random test card 2, because his program has bugs in his smithy and village function. And I checked my coverage, all the random test coverage is up to around 100%, And then I used my test dominion to test his whole program, after 100 time test, and I just checked the coverage, my coverage was up to 70%. After I communicate with him, I just made sure he has three bugs, 1; he has a bug in village. 2 He made a bug in smithy.

Conclusion:

After running the entire test, I found 4 bugs total, 2 bugs in each of dominion programs. I used various ways to detect the exact place of their bugs.

Firstly, I made some modification for some of the functions where bugs exist.

Secondly, I made some break point in some loop, and then I print out the result step by step in my test program. Thirdly, I made a lot print out and print out the result for my test, not just the final result. And then I could find the exact place. Fourthly, compared with the print result and then I compared with my dominion program, I found more than 3 bugs.

After all the test, compared with these two classmates test, I found JiaWei's program has higher coverage. And both of them I did not find any crash or stuck during my test. What's more, unit test and random test are two tests ways I like to use, these are great tools to test bugs, and then I found print is another way to know the exact bug place. Because I think print is a way which is easier than the GDB, we could just break up the program in any where, and print out the current data or print out any data in our test program. It will be a easier way for us to tell bugs.

All in all, in this term, I learned a lot for testing and debugging. This will be helpful for my future program life. I learned even how to write code is important, but we cannot make sure no one will make mistake, so know how to debug and find out bug is one of our another mission in our programming life. And I appreciate I have great professor and TAs to helped me solved a lot problems, and let me learned a lot. This is one of my best classes I have taken.