Module3, Assignment 3(reflection)

Woohyuk Yang
yangwo@oregonstate.edu

Understanding

To test my program I asked my cousin to use my program and he typed some random real number instead of a integer and the program did not work as I intended. Although this is an assignment and the instructor said " You can assume they will enter a number >= 1.", but to make this program more perfect and user friendly I thought I need to make the user can understand the intention of the programmer more efficiently and clearly and I need to write a program in a way it can react in any situation like printing out "You should only input a integer !" as a prompt to come out when the user inputs something other than an integer making users understand a way to use the program better.

Testing Plan

Description of each tests		Expected Values		results	
Test 1 The number of integers to be entered = 1 - purpose : Program correctly handles the result even if only one integer is entered in 3 different types. (0, negative number, positive number)	entered integer(s)	Min	Max	Min	Max
	-1	-1	-1	-1	-1
	0	0	0	0	0
	1	1	1	1	1

- My program could handle 0, negative number, and positive number. Test 1 went well.

Test 2	0	0	0	0	0	0
The number of integers to be entered = 2	1	1	1	1	1	1
- purpose	-1	-1	-1	-1	-1	-1
: Program correctly handles the result even if	-1	1	-1	1	-1	1
1) same integers get entered.	0	1	0	1	0	1
2) even if two different kinds of integers get entered. (negative num, positive num) (0, positive num) (0, negative num)	0	-1	-1	0	-1	0

- My program could handle when same two integers got entered.

Test 3	0	1	-1	-1	1	error	error
	0	-1	1	-1	1	error	error
The number of integers to be entered = 3	0	1	-1	-1	1	-1	1
- purpose	0	-1	1	-1	1	-1	1
: Program correctly handles the result even if	1	-1	0	-1	1	-1	1
1) integers get entered in a different order.	1	0	-1	-1	1	-1	1
2) different kinds of integers get entered in a different order.	-1	0	1	-1	1	-1	1
	-1	1	0	-1	1	-1	1

- My program could deal with two integers before. But when I tested with three different integers, to get the answer I had to enter one more integer and the answer was not still correct. I had to change the position of the first 'cin' statement to get the integer to start compare the first integer with the following integer to be entered. I found as two 'cin' statements were in the loop, two integers got compared with the next set of two integers not one by one. I had to extract the first 'cin' statement out of the 'for' loop. The comparison still could go through comparing the first integer(outside the loop) and the second integer(inside the loop). At third trial(trying the first trial again), I could find out the logic works correctly.

After altering the code, the program could handle any changes of the order of the 3 different integers. I could conclude the program does not get influenced by a different order anymore. But I did not plan to test my program with values at extreme ends of the 'int' valid range before. Reflecting the program, I tested two extreme numbers and It did work as well. I could have planned the test more complete if I had planned to test two extreme integers which could be stored in 'int' data type.

Design (Pseudocode for 3.a)

- My original design

```
set numOfInteger, minNum, maxNum to 0

get the number of integers the user wants to enter

set numOfInteger to the number of integers the user wants to enter.

Let the user type the integers as many as numOfInteger
set minNum and maxNum to the number the user first entered.

For all the integers the user would type in
  if minNum is bigger than the integer just typed
    set minNum to the integer just typed by the user

  if maxNum is smaller than the integer just typed
    set maxNum to the integer just typed by the user

print minNum and maxNum respectively
```

My design was made in a pseudocode but I did not make sure to point out the position of the 'cin' statements wich were set to get values from the user. When I finished my implementation for the first time and I tested the program, I found out I should not have positioned those statements in a way I did before. I had to extract the first 'cin' statement which would get the integer entered by the user out of the loop statement. So I could pass my test with my program.

implementation

Event though I thought I had already designed perfectly using pseudocode, when I started to implement using c++ which I only started to learn few weeks ago, I confronted some confusion which is about c++ programing language syntax. Some syntaxes I am not familiar with kept distracting me from focusing on writing the code(imlemeting this program).

For example to set a variable to store an integer which would be entered by the user, I wrote a code like below for the firs time.

```
for(int I = 1, I < numOfInteger, I++)

  int num-i = 0;
```

It was totally wrong and I catched the error reading the textbook again. To make sure the program works without any syntax errors or logic errors, I looked for some chapters which I have ever read before and seemed related to the problem I am trying to solve and read those chapters from a to z. I read some parts of "c++ prime" written by stepthen prata and used that book as a reference also after reading the textbook for the class. I visited stackoverflow.com to figure out people have same or similar problems. I could also find some information from blogs run by c++ programmers too, which was little bit distracting and vague.

What I underestimated before was a typo. I spent some time to find a cause of the error when a compiler error happened and I found it was a typo which caused the error. some typos which could happen often should be concerned, which are like semicolon, some small typos for the names of variables.

improvement

Through solving the problem I found the position of ‘cin’ for the first integer should be different from the other integers the user entered. As a ‘for’ loop goes on, input and output objects affect the program thoroughly and the positions of them is really important. ‘For’ loop should be used at the right place with other elements composing the program.

I used the text book mostly as a reference solving this problem. Many examples and sample source codes from the textbook enabled me make a structure for the program clearly and whenever encountering the difficulties which are comes to the programming language, reading a text book can be really useful. I could find information about the syntax from the books and catch typos just reading the code thoroughly, I could not have found my logical error with my design logic without the tests.

What we should do before typing in front of a computer is to think and write requirements of the problem suggested as an assignment. Like we need a nice blueprint to build a house we need to design our programs before starting implementation. I still found some difficulties implementing the program but I could solve the problems with reading books and searching for the examples on the various websites. But I have never been able to write the program just reading a book and surfing on the websites without the design process. What I can value most from this assignment3 process is not just the fact that a ‘for’ loop should be used with some concerns but I have to set some plans and follow the process of problem solving suggested before. I could find a problem with my logic during my test and I could fix it. I would not have been able to find the problem without any tests not following the problem solving process.