Personal Reflections

After I go through the SPI course this semester I found my program skills are improved so much in several areas.

1. Code Format
   Before I have this course, I just write code in a no-format format. It causes so much trouble to me when there is an error occurred. I need spend a lot of time to find out where it is and the no-format will add more trouble to that. For example, the first time I write the program01, it have no format and I cannot find where does the loop start and where it ends. What is more, it is also hard to understand what does the variable like X means. Using a good format to code, we can see the structure of the program and see the meaning of each variable, this make the program readable to the developer himself and other developers. This will also help to maintain the software if we are working in industry. It will also help developer to correct the errors and tester to find the errors.
2. Code Structure
   Code Structure is another thing I improved a lot. Write code not just in only one function but separate it in several different function. In the main function, only a few line can show the whole structure of the program. By writing like this, when there is an error happens, we can test function by function instead of looking over the whole program and try to find out where the problem is. A good code structure will also help reuse the code.
3. Code Reuse
   After get used to use a good code format and code structure, I found that it is really convenience that we can use the code we write before. Because using the structured code separate in several function that handle different problem, we can use some function without write it one more time. For example, I write a function T in program05 to find p with x, in program06 I can reuse this function T to find x with p. Another example is that the tokes.java we will use in every program. After these 10 program, I found when we are doing a big project, there will be a lot of code that will be used over and over again. Reuse the code will save effort in repeating programming.
4. Design
   Design is another thing I learn from this course. Never have design before coding before I learn this course. I have face several time that after I coding half of the program I found out the algorithm I though before is wrong. After doing design before coding, the algorithm is considered carefully to make sure this problem won't happen. Design also help to make the code in a good structure and make the algorithm more clear to developer and this will make them make less mistakes.
5. Testing
   Testing is another area I improve a lot. Before having this course I use Eclipse IDE help me to debug. The debugging tools is so powerful that make me not afraid to make mistake including mistype, overflow and forever loop. But without this tool I found the program cannot even compile. Too much mistakes in the program when

I first not using IDE. What is more, without using the debug tools, I need to write some output to know what happened in a particular line and write some testing code and input example to test if every function works well. Now I am used to test the program function by function to find out where the problem is.

In conclusion, PSP helps me a lot in programming, it improve not only the productivity but also the quality.