Project 1: Milestone 1
MIPS Assembly Language Programing - II

Team 6:
Gaurav Mishra 201202057
Mayank Agrawal 201303003
Aakar Gupta 201250877

Bugs Detected:

1) When we open a file it comes up with name mips(number).asm, where the number keeps on increasing every time even if we close the previous windows. (The new file should be named as untitled.asm) and a user should be asked to name it.
2) Select all option for selecting the entire file does not work when applied on the already given files but it work on the new files.
3) The error in assembling the code does not give the line number. Like when someone uses some unidentified instruction.
4) Copy option for copying the selected code does not work when applied on the already given files but it work on the new files.
5) There is no documentation on MIPS procedure calls. calls in the Help section.
6) Undo functionality is not working on new files.

Test Cases:

We have developed few test cases that will help us in rectifying above bugs and will help us maintaining a correct state of the final code. (Things working fine now should also work the same way after Refactoring). These test cases can be viewed here.
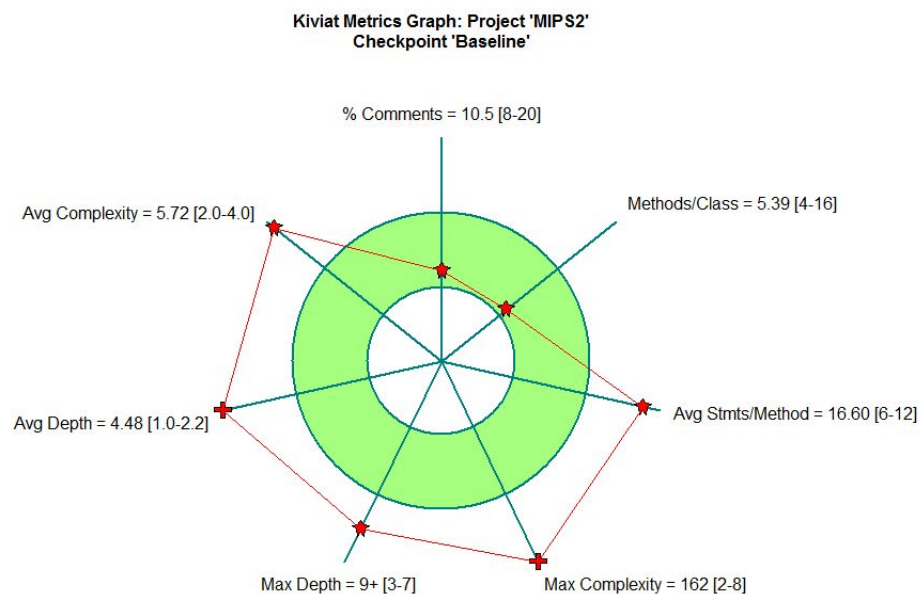
Quality Metrics we will be evaluating:
1) Abstractness
2) Average Block Depth
3) Average Cyclomatic Complexity
4) Average Lines Of Code Per Method
5) Average Number of Constructors Per Type
6) Average Number of Fields Per Type
7) Average Number of Methods Per Type
8) Average Number of Parameters
9) Comments Ratio

10) Efferent Couplings
11) Lines of Code
12) Number of Characters
13) Number of Comments
14) Number of Constructors
15) Number of Fields
16) Number of Lines
17) Number of Methods
18) Number of Packages
19) Number of Semicolons
20) Number of Types
21) Weighted Methods

Currently we have evaluated the above matrices using **Codepro** and our results are documented here

The **Source Monitor Metrics** for particular methods are as follow :

Metrics Summary For Checkpoint 'Baseline':

Major details are:

Project Name	MIPS2
Checkpoint Name	Baseline
Created On	1 Feb 2016, 22:12:40
Files	7
Lines	3,221*
Statements	2,247

Percent Branch Statements   19.7
Classes and Interfaces   23
Methods per Class   5.39
Average Statements per Method   16.60
Maximum Complexity   162

Timeline:

Milestones	Expected Finish Date
Understand existing Java code base	02-Feb-2016
Figuring out current design patterns and antipatterns [Re Engineering]	03-Feb-2016
Conversion to Javascript	09-Feb-2016
Updated test cases (functional and nonfunctional)	11-Feb-2016
Updated Quality Metric for Javascript	12-Feb-2016
Refactoring javascript code	15-Feb-2016