Test cases

mandag 7. mars 2016 12.46

Requirement 1
The assignment we chose to work on was the second one, Minesweeper-Java, as Java is the language we feel most comfortable around when it comes to coding. The Minesweeper program is a basic console printing application that lets you play the classic game Mineswee per. We are using Atom/Sublime to modify the project.

Testing the functionality of the program. We are using test-cases to specify what parts of the program we want to test. This is to ensure that the program works as it is specified in the requirements of the application(we assume that it's supposed to be flawless and work just like any other minesweeper game).

#	Test-case	input	Expected result	Actual result
1	Testing integers outside the row/column domain to see that it does not accept such values	5 10 -1 -1 -1 20	"Invalid input" "Invalid input" "Invalid input"	"Invalid input" "Invalid input" "Invalid input"
2	Testing integers for slots that you've already revealed, to see that revealed slots cannot be replaced.	0 0 0 0	Correct or boom/end "You stepped in already revealed area!"	Correct "You stepped in allready revealed area!" Spelling error revealed!
3	Make sure that "Invalid input" does not count as a turn.	Enter 4 5 4 5 4 5	Turn = 0 Turn = 1 Turn = 1 Turn = 1	Pressed enter without any input multiple times, this did not affect the turn integer Tried input row 4 column 5 several times, this did not affect the turn integer
4	Make sure that when you type "exit" or "restart" with 0 turns, the program will exit without registering your name and score as this is useless.	Exit Restart	End Restart	The program registered our score even though we put no effort into obtaining a score when exiting/restarting, this should be changed in the code
5	Testing "top"-command with no results	Top	"Still no results"	"Still no results"
6	Does the program end correctly when you've finished the board?	All slots	"Congratulations you WON the game!"	//Commented out the function "boom();" at line 145 in MineField.java, allowing us to finish the board without it ending from a boom :) We won the game FAIRLY and it showed with the correct score of 35. Result: "Congratulations you WON the game!"

We have decided not to perform any non-functional tests on the source code. The program is user friendly if you know basic DOS, it is hard to break at any point with e.g. spamming the enter key and such until it breaks. However we would like to bring up one non-functional issue that we found when we tried modifying the board size
The colum numbers are hard-coded when printed, and fixing the printing causes the column-line to look ugly. This is something we only played around with, so we only wanted to mention it here.

Our previous knowledge is is minscule and therefore we struggled to come up with ideas for testing in this case.