Bug Report For Team X

General Methods for Testing

1. Run our and tdd test cases
2. Edit test cases to cater your implementation
3. If still not pass, run debug mode and try to find the cause
4. Open an issue to record every bug

Bug Report Table

Bug#	Description	Testcase `(test.hackathon.*)`	Failing Code	Comments

1	The bug is due to invalid handling of: Project Description 8.6 Globbing
	For each argument ARG in a shell command that contains unquoted * (asterisk) do the following:
	3. If there are such paths, replace ARG with a list of these paths separated by spaces.
		`GlobbingTest.`
		`testGlobbing()`
			`RegexArgument.`
			`globFiles()`	

2	This bug is due to invalid to distinguishing relative path and absolute path in Windows OS	`LsApplicationTest.`
		`runWithValidArgs()`
			`LsApplication.`
			`resolvePath()`	

3	The bug is due to invalid handling of: “Cuts out selected portions of each line (as specified by list) from each file and writes them to the standard output”
	Ref. spec page 18, Section 9.11
		`CutApplicationTest.`
		`testRunWithTwoFile()`
			`CutApplication.`
			`cutFromFiles()`	

	The implementation fails to cut file with multiple lines			
4	The bug is due to invalid handling `diff` a file with InputStream that contains spaces using `-B` flag. Ref. spec page 15, Section 9.6, No assumption mentioned. When diff between two files using `-B` flag, like `diff -B FILE1 FILE2`, it works correctly.	`DiffApplicationTest.testRunWithStdinAndFlag()`	`DiffApplication.diffTwoFIles()`	
5	The bug is just a miss in output display requirement when diff two directories. Ref. spec page 15, Section 9.6, No assumption mentioned. Unlike missing NEW_LINE or spaces, this miss is considered as an incomplete implementation after our discussion.	`DiffApplicationTest.testDiffDirContainFilesWithDifferentContent()`	`DiffApplication.diffTwoDir()`	
6	This bug is due to wrongly ignoring std_in with all NEW_LINEs Ref. spec page 14, Section 9.3, No assumption mentioned. `paste /path/to/file` should get exactly the same result with `paste - < /path/to/file` But the outputs of those two commands are different.	`PasteApplicationTest.testMergeStdin_multipleLine()`	`PasteApplication.mergeStdin()`	
7	This bug is due to invalid handling cp an existing file to a non-existing folder. Ref. spec page 17, Section 9.10. No assumption mentioned. Taking an example command `cp *.txt foo` as an example, if foo is not exist, your CpApplication would treat foo as a file rather than a folder, which does	`CpApplicationTest.testCpToNotExistFolder()`	`CpApplication.cpFilesToFolder()`	

	meet the specification `cp [FILES] DIRECTORY`			