

Grading Instructions

Total Points: 60

- gzip has been created correctly (no executables, .class, backup file, or jar files): 5 points
- Code compiles and runs according to the instructions in the assignment (which should be the same in the README.md): 10 points.
- Creating a Github account and forking the assignment (using the link provided) before 11:59 PM EST, June 3rd: 5 points
- Correct user of git
 - 2 points: at least 6 commits in total.
 - 3 points: 5 substantial commits (at least 75% of the code) 72 hours before the submission deadline.
 - 3 points: meaningful commit messages for each commit.
 - If more than 50% of the number of lines of code is committed in the last 24 hours: -5 points
 - If more than 50% of the number of lines of code is committed in the last 48 hours: -2.5 points
- Code generates correct output file: 5 points
- Correct value for AVG_NUM_WORDS_PER_SENTENCE metric: 3 points
- Correct value for AVG_WORD_LENGTH metric: 3 points
- Documentation for each method (One line explanation + return type) using javadoc style: 10 points
- Boundary conditions checks
 - Empty input file: 2 points.
 - Missing input file: 2 points.
 - Words contain characters other than [a-zA-Z0-9]: 2 points.
 - Empty line in input file: 2 points.
 - Missing period character in a line: 1 points.
 - More than one sentence per line: 2 points.
- Penalty points for input processing.
 - Reading in the entire input and storing the strings in a data-structure: -2 points
 - Performing multiple reads of the input file: -2 points
 - Using a library function for rotation of words: -2 points
 - Using the built-in average calculator when calculating metrics: -2 points
- Penalty for any exception that is not handled: -5 points each time

Back to [Programming Design Patterns: Assignment 1](https://sites.google.com/binghamton.edu/csx42-summer-2020/assignments/assign1/grading-instructions.html)