

Assignment 4: Adventure Part 2 Code Review Rubric

This rubric is a set of guidelines on what we are looking for in each area. The check boxes should not be thought of as points of equal weight but topics to think of when working on your assignment.

Scaling Factor

All assignments will have the following scaling factors. These will be applied to the final grade for the assignment rather than on each section.

- Lose 25% for each day late
- Final score will be scaled by proportion of work complete
- Lose Percentage of material taken from sources if over 25% not counting recommended libraries and provided code

Improvement (20%)

- Improved Amazing Adventures code using feedback given in code review and on Gradescope*
- Effectively used original Amazing Adventures code to complete Amazing Adventures Part 2*

Object Decomposition (15%)

- Purpose of each class/object is clear and well thought out
- Classes have good cohesion (related things together) and loose coupling (unrelated things separated)*
- Classes are at the appropriate granularity (not too big and not too small)
- Functions are appropriately static (if they don't operate on an object) or non-static (if they do)

Overall Design (15%)

- Solution approach is well thought out
- Code is logically organized
- Avoids unnecessary repetition ("Don't repeat yourself")

Documentation and Commenting (10%)

- Comments for non-trivial public methods
- Appropriate use of comments (summaries, commentary, big picture, likely questions)
- Comments provide information that can't be quickly learned from the code itself

Naming and Layout (5%)

- Names succinctly and accurately describe the named entity
- Braces, indentation, line length/wrapping
- Horizontal and vertical whitespace group related, separate unrelated things
- Naming and layout meet (Google Java) coding style guidelines

Automatic Testing (15%)

- Important classes of inputs are tested (valid, invalid/errors, boundary)*
- Tests well documented through naming (or comments if necessary)
- Tests are well-organized (logical grouping/order, generally one assertion per test)

Process (10%)

- Code was checked-in periodically/progressively in logical chunks*
- Meaningful commit messages
- Effective reuse of code from last week

Presentation (5%)

- Arrived on time with all necessary materials and ready to go
- Good selection of topics to focus on and logical order of presentation
- Appropriate pacing and engagement of the fellow students
- Speaking loud enough and enunciating clearly

Participation (5%)

- Engaged and paying attention to other students presentations
- Asks questions and/or makes comments that further the discussion*
- Explains reasoning for why something is good or bad
- Behaves respectfully to moderator and other students

Extra Credit