

# CS106/206 Assignment HC5/BMC6 Style Grading Rubrics

## General

7 points are allocated to fairly mechanical rules on naming/comments/indentation - these should be easy to check off. Another **18** points are allocated to more creative practices, as explained below. Consult the formatting guide for details to check for under each category

Print student programs from Emacs, via “postscript print buffer” menu option.

### Total: 25 points

#### Code formatting (7 points total)

1. Naming Conventions: **2 points**
  - a. if any of the rules are violated
2. Whitespace: **1 point**
  - a. inconsistent spacing (excessively) - - if just one place, point it out but don't take off
3. Comments: **3 points**
  - a. File header missing or malformed
  - b. Uncommented instance variables - no comment is okay if well-named
  - c. Uncommented methods (getters and setters can have no comments, when appropriately named)
  - d. Method comments that do not conform to javadoc style
  - e. Uncommented complex blocks of code
  - f. Unhelpful comments
4. Indentation: **1 point**
  - a. inconsistent indentation (excessively) - if just one single line, point it out but don't take off

#### Design principles (18 points total)

The exact point allocations will change from assignment to assignment. In general, because it is impossible for me to imagine all the ways thing can go wrong, grade somewhat holistically instead of sticking to the rubric strictly.

#### Assignment 5/6 (binary tree)

1. private instance variables and getters **1 point**
  - a. Any non-private instance variables, including missing modifier
2. Use `public/private static final` constants instead of `integer/double/String` literals - any literal that has reason to be changed later should be a constant **1 point**
  - a. Using `[0]`, `[1]`, `[2]` ... directly in code after calling `split` (does not apply to Haverford, which uses `CSVReader`)
3. Constructor must initialize all instance variables **1 point**
4. `LinkedList` (**12 points**) (note that `remove` is extra credit for HC)

- a. private nested Node class
  - b. insert, contain, remove, toString\*Order implemented recursively. If not, -3 each to a max penalty of -10
  - c. insert, contain and remove does comparisons via .compareTo
  - d. toString overridden appropriately
  - e. No additional data structures (take off 3-5 points depending on how bad it is)
5. Custom class say Candidate for polling data (3 points)
- a. Implements Comparable<Candidate>
    - i. if implements Comparable and compareTo casts Object to Candidate instead, don't take off but write comment that Comparable<Candidate> is better
  - b. toString overridden appropriately