

Prescribed Coding Standards for the Submission

- **Operators:**

- When defining expressions using operators, there should be a single space before and after the operator. Operations can be arithmetic (+, -, *, /, % etc.), relational (>, >=, <, <=, !=, etc.), conditional (&&, ||, etc.) or assignment (=).

- **Variables:**

- Fields must only be initialised in the constructor, unless they are static or final.
- Local variables must be initialised when they are declared.
- The use of public fields must be justified.
- Variable names must not contain an underscore (_) (except final variables).
- Final variables must be in all caps and must use an underscore to split up multiple words in the name.

- **Methods:**

- Methods must have exactly one blank line between each method, and between the start and end of the class's delimiter ({ })'s.
- Method names must be or contain action verbs. Names should be descriptive and convey a high level idea of the purpose of the method.
- Method names must use camel casing with the first character always lower case.
- Methods must be sorted alphabetically (except constructors, which always appear first).

- **Classes:**

- Class names must be nouns and must be singular.
- Class names must follow camel casing with the first character always upper case.
- Each file must only contain one class definition.
- A complete class template must include:
 - one default constructor
 - at least one non-default constructor (if the class has fields)
 - accessors for each field
 - mutators for each field
 - a display/toString method.

- **Brackets and indentation:**

- All code within a specific block must have exactly one indent. One indent is equal to one tab or four spaces.
- The placement of { }'s to start and end a block must appear on their own line. Opening braces ({) especially should not be placed on the same line as the header.

- **Code:**

- Access modifiers are not optional and must be clearly defined for methods and fields.
- The maximum code line length should be 80 characters. To prevent code going off the screen, consider breaking the code across multiple lines.

- **Documentation:**

- All classes and methods must use correct documentation.
- Inline comments are not necessary unless required to explain a key concept. Excessive inline comments must be avoided as they affect code readability.