**Dream Team Coding Standards**

# **Naming Variables**

## Variables

* Variable names should use camel case, single worded variables should be lowercase
* Avoid the use of single lettered variables, unless for use in iteration
* Variables should be named according to the value or object it represents
* Prefixes should be used for Boolean variables, isBool or hasValue
* Postfixes should be used for counting variables, numCount

Example:

int exampleVariable;

int example;

## Constants

* Constant names should be in all capital letters, with spaces represented by an underscore

Example:

const int EXAMPLE\_CONST = 0;

## Classes

* Class names should use upper camel case, with each word beginning with a capital letter, no spaces or underscores

Example:

class ExampleClass{

//classy stuff

}

# **Naming Functions**

## Functions

* The same naming conventions for variables should be used for naming functions, camel case
* Functions should be named according to its job, with a descriptive verb (get, do, etc.)

Example:

getValue(){

//function stuff

}

# **Program Flow**

## Conditional Statements

* Conditional statements and parentheses should be separated by a space
* Iterative loops over numbers should use single letter variable names such as i, j, k, m
* Iterative loops over objects should use ‘it’
* Multiple conditions should be on separate lines, with the conditional operator starting each new line

If Statement Example:

if( condition ){

// statements

}else if( condition ){

// other statements

}else{

// more statements

}

While Loop Example:

While( condition ){

// statements

}

For Loop Example:

for( int i=0; i<MAX; i++ ){

// statements

}

Long Conditional Statement Example:

if( condition1

|| condition2

|| condition3 ){

// statements

}

## Indentation

* Write only one statement per line
* Write only one declaration per line
* Add at least one blank line between method definitions and property definitions
* Use parentheses to make clauses in an expression apparent
* one tab (equivalent to 4 spaces) should be used for each level of indentation

Example:

if ((val1 > val2) && (val1 > val3)){

// Take appropriate action.

}

## Comments

* Place comments on separate lines, not at the end of a line of code
* Begin comment with an uppercase letter
* End comment with a period
* Insert one space between the comment delimiter (//) and the comment text
* No formatted blocks of asterisks around comments
* In every header file, comment should be placed containing
  + Name of the file
  + Name of author/contributor
  + Additional information about its header file

Example:

// Here is a proper comment.

// Here is another one.

/\* main.h

Joe Vandal \*/