College of San Mateo
CIS 117 Python Programming

Coding Standard for our Lab Assignments

When you convert a design document into Python source code, one of your primary goals should be to write the source code and internal documentation in such a way so that it is easy to verify that the source code conforms to the design, and so that it is easy to debug, test, and maintain the source code. The coding standard below is designed to assist you in achieving this goal.

1. Source Code Format

Source code lines will be no more than 79 characters long. Break long lines into multiple shorter lines using the continuation character ("\").

Only one statement will be included on a given line of the source code.

Blank spaces and blank lines will be used to enhance the readability of both source statements and comments.

Use four spaces for indentation.

Control structures will not be nested too deeply. If the clarity of the source code is impaired by too many levels of nesting, the statements that are nested most deeply will be removed and placed in a separate function.

2. Mnemonic Identifiers

Meaningful identifiers that are taken from the problem domain will be used for all names. Furthermore, a name will be used for one purpose (the same name will not be used for more than one purpose).

3. Symbolic Constants

Symbolic constants will be used instead of embedding arbitrary numeric and character constants in the source code. Use "upper with under" style for all symbolic constants (see below).

```
HEAT_OF_FUSION = 79.71 # Calories to melt one gram of ice
```

4. Variable Names

Use "lower with under" style for all variable names. In some situations, it is appropriate to append type information to variable names as a visual reminder about the purpose of those variables:

```
student_count = 0 # Number of students in class
fahrenheit_float = 0.0 # Temperature in Fahrenheit
```

5. Use “CamelCase” style for all class names:

```
class SalariedEmployee
```

6. Descriptive Comments

The source code will include comments that describe the functionality of significant blocks of code. Comments are particularly important before all blocks of code that perform major data manipulations or error processing.

7. Source Code Header

Each source code file will contain an introductory block of comments that identifies the assignment and gives a brief overview of the program. For example:

```
#############################################################
# CIS 117 Python Programming:  Lab #1
#
# Compiler, Tabs, Indentation, User Output
#
#############################################################
```

8. Guido's Style Guide

Guido van Rossum, the creator of Python, has a lengthy style guide:

http://www.python.org/dev/peps/pep-0008/

This style guide is an excellent resource for more complex situations not described in this document.