

# CS 1005/2005 Documentation Guidelines

Documentation for any program falls into two categories: internal and external. **Internal documentation** consists of comments included in the program, use of meaningful identifiers, and consistent and proper use of indentation and spacing that helps convey the structure and meaning of the code. **External documentation** is typically written as a document separate from the program itself; it usually consists of a user guide and sometimes includes a detailed description of the design and implementation features of the program. For this course, you will be expected to follow the following documentation standards:

## Internal Documentation

### Comments

1. Each program should begin with an introductory comment of the following form:

```
/*
 * <filename> - <brief title and/or description of program>
 *
 * written by:  <your name>
 *              <wpi login name>
 *              <your section number>
 *              <date>
 */
```

2. Each function should begin with a comment of the following form:

```
/*
 * <function name>
 *
 * PRE:  <list of pre-conditions, usually stated in terms of input arguments>
 * POST: <list of post-conditions, usually stated in terms of output arguments>
 *
 */
```

**pre-conditions** are conditions that must hold prior to the invocation of the function. **post-conditions** are those conditions that are true after the function returns.

3. There should be a comment next to each declaration in the program, unless the identifier sufficiently conveys the intended use of the declaration.
4. Obscure or complicated sections of code should be *preceded* by a comment which explains the intent of the code.

### Identifiers

1. Variable names should convey the intended meaning of the variable. For example, *found* might refer to a variable that contains a Boolean value, which gets set to *true* when a searched-for item is detected, and is *false* otherwise. If an identifier consists of more than one English word, capitalize each new word (*currentTime*).
2. Function names should be verbs that indicate the action performed by the function, for example *InitializeArray()*. Capitalize each word in the function name. Functions that return Boolean values should be named with verb phrases that describe the condition that holds when the function returns the value *true*, for example, *IsFull()*.

### Indentation and Spacing

1. Align all statements that appear at the same level. Use three or four spaces for each nested level of indentation (be consistent).
2. Align a comment that describes a block of code with the code. Precede the comment with a blank line.
3. Place the curly braces that delimit the body of a function in column one, and on lines by themselves.
4. For compound statements, place the opening brace on the line that begins the compound statement and place the closing brace on a line by itself, lined up with the line beginning the compound statement, for example,

```
if (i == 0) {  
    done = true;  
    j++;  
}
```

## External Documentation

You will be expected to submit an external documentation file along with each programming assignment you do this term. The format to be followed can be found in the description of the [Computer Science Department Documentation Standard](#). You are to complete the following sections of the documentation file for every homework assignment:

Author (your name)  
Date  
Version (1.0)  
Project ID (HWx)  
CS Class (CS 1005)  
Programming Language  
Problem Description  
Program Assumptions and Restrictions  
Interfaces (just the "User" section)  
Implementation Details (just the "Variables" section)  
How to build the program  
Additional Files  
Test Procedures  
Performance Evaluation (include only if you have special concerns or wish to point out features of the program)  
References

Note that the sections on Overall Design and Implementation Details need not be completed; for the most part, these will be given to you as part of the assignment. The program source code and Results (script file) will be turned in as separate files, and need not be included in the documentation file. Any sections of the documentation file that are not completed should be marked N/A for not applicable.