Part 1 – External (Report)

Full size, professional programs require at least three documents. These are the *Program Design Document*, the *User's Manual* and the *Technical Reference Manual*. In this course, you will combine these documents into a single cover memo, since your programs are relatively short and simple. Address your memos to me as your "Engineering Manager at Acme Engineering." Double space your memo (except for tables and source code) to give me room to write comments. Submit your memo and source code via e-mail. Your memo should generally be no more than 3 pages long and contain the following short sections (section headings not required):

Introduction

In this section, describes what the program does. The introduction should essentially restate the assigned problem and layout specifications for the work to be done.

Operating Instructions

In this section, explain how to use (not compile!) the program. This section serves as the User's Manual.

Input/Output Description

In this section, describe the input required by the program (if any) and the output it produces. This section should contain a clear statement regarding whether the program operates in batch or interactive mode. If the program is interactive, this section should indicate whether the user interface is graphical or text/console based. It should include descriptions of all values that must be entered during program execution, the format of any input and output data files (not your source files) and the units of all values having units.

Program Design/Algorithm

In this section, list the classes used and how they interact. For each user defined class used, the purpose for class should be described (with emphasis on what it represents), the type and meaning of each member variable should be listed and the function of each method should be described. For methods involving more than one selection or looping construct there should be a prose description plus pseudo-code **or** a flowchart (both are not required, but can be included) of how it works. This section serves as the Design and Technical Reference documents.

Sample Solutions and Test Results

These are solutions to the problem using some method other than the computer program. They are often done by hand, but could include experimental or other results. Sometimes they are solutions for some special cases of the problem that are particularly easy (or at least possible) to solve by hand. These results should be **explicitly compared** to those from the program to demonstrate that the program works. This material is often best presented in tabular format. The number of specific cases done should be sufficient to prove beyond a reasonable doubt that the program works.

Appendices (as appropriate)

Attach to e-mail and/or include in archive (.zip) file.

Source Code Listing

The complete source code (.m) files. Remember to include plenty of comments (see below) in your source code.

Screen Shots (if appropriate)

Screen shots of the program in operation with explanatory text. These are particularly important for GUI programs but may be useful for console mode programs. Screen output captured from your PC (use Fn PrtSc on most laptops)

Data Listings (if any)

Input and output data files as appropriate.

Literature Sources (if any)

Citations (references) to any literature used in the design and development of the program. This could include class handouts, websites (like the Medical Algorithms site, www.medal.org), textbooks, professional journal articles, etc.

Your documentation should be neatly prepared and submitted as a single Word (.doc) document named according to my standard naming convention (generally your MSOE user name followed by the assignment number) and attached to an e-mail message sent to tritt@msoe.edu.

Part 2 – Internal (for Matlab **help** and **lookfor** commands)

In general, files should begin with a block of comments for use by the Matlab help system. These should include a header (**lookfor**) line, purpose, pre and post conditions, inputs, outputs, variables and the revision history. See my sample program and function handouts (Set 0) for format templates and examples of appropriate file comments.

Part 3 – Internal (general purpose)

Other Comments - Your program must contain enough comments to make its operation easy to understand. This doesn't mean every line should be commented, just each major section or block of code.

Other Internal Documentation - A logical program design, consistent indentation and mnemonic variable, class and function names go a long way toward creating a well documented a program and will be considered in the grading.