1   Programming Assignments

Correctness is an important criterion, but does not guarantee a top mark. Grades on programs will be based on:

1. Correct behavior on normal or typical input; adherence to specifications.

2. Robustness: Correct behavior in extreme or unusual situations; reasonable recovery from unusual or incorrect inputs or from internal bugs.

3. Readability: comments, mnemonic identifiers, clear structure, indentation, external documentation. Every program should have external documentation telling how to use the program and what its limitations are. It should also have internal documentation that describes data structures and algorithms. Every procedure should be documented to show what it does and what its parameters mean.

4. Quality of test data: tests that thoroughly exercise the program and explore unusual cases. You are expected to provide your own test data for every program as well as demonstrate its accuracy on standard data. We will run your programs on our own test data, which will not generally be made available to you in advance. We encourage students to share test data with each another.

5. Efficiency: avoiding unnecessarily inefficient algorithms or constructs. However, efficiency should never be pursued at the expense of clarity.

2   Documentation

Documentation for any program falls into two categories: internal and external. **Internal documentation** consists of the comments included in the program. They generally fall in three places: at the heads of procedures, at declarations, and at particularly tricky or opaque segments of code. Procedure-head comments describe the effects of the procedure and assumptions about its inputs and outputs. The procedure name should convey as much of this information as possible. Names of variables, types, constants, and field selectors should also be as meaningful as possible, with a comment next to the declaration to fill in extra information. Be as specific as possible: Say "index of last element on the stack" rather than "top of stack". Such comments should be written *before* the rest of the program. Your code should be as clear and simple as possible, but occasionally you may need to resort

to a clever trick. In such cases, a comment will help the reader. For example, "This loop rearranges a[i..j] so that elements less than x precede those greater than or equal to x".

External documentation (which you may include as a long comment right at the beginning of the program itself) has two types: for the typical user, and for someone who wants to understand how the program works. The first kind of documentation includes details of how to call the program, its options, formats of data, limitations, bugs, and special features. For course programs, part of this information is included in the assignment and need not be repeated. Emphasize both the negative aspects of your program (limitations, known bugs) and the positive aspects (extensions, special features). If you don't include the former, we will assume you didn't realize the limitations were there. If you don't include the latter, you may not get credit for special features.

The other kind of external documentation is an overview of the algorithm used in the program; parts of this information included in the assignment need not be repeated. Include anything that may be different from what other students in the class did, or that is not immediately obvious. One goal of this documentation is to give the reader an overview of how the program works, so that they can understand smaller pieces of it without having to understand each procedure.

Make sure you provide a comment at the beginning of each source file describing the purpose code. Also place a comment at the beginning of each C/C++ procedure/method to both describe the procedure/method and to make it easier to find when scanning the file. You may want to include additional information as specified in the Computer Science Department Documentation Form available at the `http://www.cs.wpi.edu/Help/documentation-standard.html`.

3   Written Assignments

Although content is a significant part of any homework writeup, content means little if the reader must toil at extracting the information or read between the lines. Written assignments will be graded on presentation as well as content. Answers should be short, concise, follow directions, and answer questions precisely. While it is not our intent to play English teacher while grading homeworks, we will not ignore flagrant errors in grammar, organization, or style.