Table 1. Project Scoring for 30-point reports

Item	Value
Author identification: Cover Sheet and Author identification on outside cover of binding.	1
Binding (see note 1)	1
Interface description (note 2)	9
Algorithm description (note 3)	9
Test plan and results (note 4)	9
Source listings	1
Total	**30**

Notes:

1. The binder must be a folder having an inside pocket on both the front and back cover and three soft metal posts with which to secure a document. The document must have three holes punched in the left edge and bound inside the folder using the three soft metal posts. The binder must have your name, student id, course, and section number typed on the outside front cover of your folder.
2. An interface description includes:
   - 2.1. The name of the program and a description of program parameters that appear on the command line must appear in the interface description. An example of how the program is called must be part of the interface description.
   - 2.2. A brief description of the actions performed by the program must be included.
   - 2.3. If the program reads input data from a file, a description of the file and an example of the file format must be included.
   - 2.4. If the program writes data to a file, a description of the file together with an example must be included.
   - 2.5. A brief description of every member and member function of every class in your program.
   - 2.6. Examples of how your classes are used must be included in your interface description.
3. An algorithm description includes:
   - 3.1. An enumerated, step-by-step discussion of how data are transformed by your algorithm must be included in the algorithm section.
   - 3.2. Diagrams of principal data structures must be referenced in the step-by-step discussion.
   - 3.3. An example of how your algorithm transforms actual data must be included to illustrate key features of your design.
   - 3.4. Time complexity analysis.
   - 3.5. Results of empirical time tests that validate your time complexity analyisis.
4. Test plan and results:
   - 4.1. Describe how you plan to test your program.
   - 4.2. Design test data that stress your algorithms.
   - 4.3. Record all test data, predicted, and actual results in this section.