Sim01 - Grading Rubric and Assessment Form

Student Programmer Code Number: _____

Student Grader Code Number: _____

Grading annotation is **required** where lines are provided

Quality building process – must be easy to read and build

_____ / 5. Program compiles without warnings or errors; makefile is correctly configured so that typing **`make`** will implement the building process; all required code, libraries, etc complete and available

- no more than 2 points credit if unecessary files such as unused library files, program configuration or meta-data files, etc. are included

_____

_____

_____ / 35. Program source code is easily readable and understandable (with consideration for the Programming Standards document)

≤ 5 pts. Difficult or impossible to read or understand, poor indenting and program structure
≤ 15 pts. Some parts difficult to read or have poor structure, but some program parts are clear
≤ 25 pts. Some parts difficult to read or have poor structure, but overall program process is clear
≤ 35 pts. Program is written and structured clearly, all parts are quickly and easily undersood

_____

_____

_____

_____

Quality program development – specified items must be easy to identify and understand (no credit if unique threads are not used for each I/O operation, or assignment specifications not followed)

_____ / 10. Program and code are structured well; functions are appropriately used to support program modularity; code is efficient and is not repeated unnecessarily

_____

_____

_____

_____ / 5. Program responds elegantly and appropriately to configuration and meta-data input file failures, such as corrupted or missing files

_____

_____

_____

Correct program code/operation - items must be clearly found in code prior to running program (no credit for any of the following if unique threads are not used for each I/O operation, or assignment specifications not followed)

_____ / 5. Correct and appropriate thread calls and management

_____ / 5. Correct and complete management of all input data (e.g., configuration, meta-data files)

_____ / 5. Correct and appropriate use of system clock to report OS simulation times; each operational time (e.g., processing, I/O, etc.) is reasonably close relative to the given cycle time and msec per cycle data

_____ / 5. Program runs correctly with 5 meta-data sets (3 provided, and 2 grader-generated) (1 pt each)
- **no credit here** if program does not run (e.g., won't run on ECC/Linux computers, `make` file failure, code or code support files missing, etc)

_____ / 75. Assignment Subtotal

☐ **C language was used exclusively (must compile with gcc)**

Instructor grading area below. No student writing under this area

---

_____ / 75. Assignment Subtotal, less late submission reduction if appropriate

_____ / 5. Extra Credit for using C language exclusively

_____ / 25. Grader Score

_____ / 100. Total Simulator Program Grade