Sim03 - 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

_____

_____

_____ / 5. 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
≤ 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
(**any** single-letter variables: -10, more than 1 global variable or data structure: -5)

_____

_____

_____

_____

Quality program development – specified items must be easy to identify and understand (no credit for these items if unique threads are not used for each I/O operation or system does not use interrupt process)

_____ / 15. Correct and clearly understandable Round Robin scheduling strategy, and with FIFO (and SRTF for graduate/extra credit)

_____

_____

_____

_____ / 5. Correct management of input files and data along with error/exception management if data files are missing or corrupted

_____
_____
_____

_____ / 10. Correct implementation of a five state process management system, including clear and understandable use and management of Process Control Block data

_____
_____
_____

_____ / 10. Correct and clearly understandable implementation of interrupt process and process management as a result of interrupt actions

_____
_____
_____

Correct program code/operation - no credit if unique threads are not used for each I/O operation or system does not use interrupt process

_____ / 15. Program runs correctly with 5 grader-provided meta-data sets (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)
- run times of processing and I/O must be reasonably close to configured cycle times

_____
_____
_____

_____ / 100. Assignment Subtotal ☐ **C language was used exclusively**

☐ **Reasonable attempt at Graduate Requirement/Extra Credit – development of pre-emptive Shortest Remaining Time First (SRTF-P) operation that is implemented after each process completes (Check Box).**

Instructor grading area below. No student writing under this area

---

_____ / 100. Assignment Subtotal, less late submission reduction if appropriate

_____ / 10. Extra Credit Attempt (proportional to Assignment Subtotal)

_____ / 5. Extra Credit for using C language exclusively

_____ / 25. Grader Score

_____ / 125. Total PA Grade