CITS2002 Systems Programming 2020, 1st Project Marking Sheet

Submission of **HE, RUIDA (22762872)**

Marks awarded from manual code inspection:

• Clear, descriptive, and sufficient comments	2 / 2
• Use of #defines for constants, rather than embedded numbers	2 / 2
• Consistent layout and regular indentation of code *Too much line spacing.*	1 / 2
• Choice of meaningful identifier names	2 / 2
• Design of data-structures to store process, and pipe information	4 / 4
• Initialising data-structures while reading the eventfile (just once)	2 / 2
• Data-structures and functions to manage Ready queue, states, events *Separate helper functions should not be created to do the same job (eg dequeue_R() and dequeue_B())*	1 / 2
• Control-flow within, possibly new functions supporting, run_simulation() *Should not need three if / elif / else code blocks in sys_readpipe() (no bytes in the pipe to be read and more bytes that need to be read than are available should be able to be combined into one code block). (minor issue so have not removed a mark this time).* *Otherwise ok.*	4 / 4

Any other comments:
Please review the sample solution -
https://teaching.csse.uwa.edu.au/units/CITS2002/projects/project1.php

---

TOTAL MARKS FROM MANUAL CODE INSPECTION: **18 / 20**

TURN PAGE FOR MARKS AWARDED FOR TESTING PROJECT

CITS2002 Systems Programming 2020, 1st Project Marking Sheet

Submission of **HE, RUIDA (22762872)**

Marks awarded from compiling submission and running a series of tests:

• 1 syntactical or logic change required to compile or execute submission	1 / 2
• Successful - A single process exits immediately	2 / 2
• Successful - A single process computes (for less than the timequantum), then exits	2 / 2
• Successful - A single process computes (for multiple timequanta)	2 / 2
• Successful - Parent forks new child, parent and child both execute	2 / 2
• Successful - Parent forks two child processes, all execute	2 / 2
• Successful - Parent forks child process, child executes, parent waits for child	2 / 2
• Successful - Initial process sleeps, computes, sleeps, computes, and then exits	2 / 2
• Successful - Process creates pipe, forks child, parent writes to pipe, child reads from pipe	2 / 2
• Successful - Process creates pipe, forks child, parent writes to pipe, child reads 4x from pipe	2 / 2

TOTAL MARKS FROM AUTOMATED TESTING: **19 / 20**

PROJECT TOTAL: 37 / 40