COMP3520 Operating Systems Internals Assignment 2 Part 1 Marking Scheme

Source Code

Criteria	Mark /10
• Demonstrates mastery of relevant scheduling concepts • Successfully implements an optimal solution that fully meets the assignment requirements with no errors or omissions • Codes with expertise, demonstrating exemplary skills in producing correct human-readable source code that compiles on the School of Computer Science servers	10
• Skilfully applies relevant scheduling concepts • Successfully implements an optimal solution that meets the basic coding requirements with, at most, two minor errors or omissions • Demonstrates excellent skills in producing human-readable source code that compiles on the School of Computer Science servers	9
• Competently applies relevant scheduling concepts AND • Implements an optimal or near-optimal solution that meets the basic coding requirements but with a few minor errors or omissions AND • Demonstrates excellent skills in producing human-readable source code that compiles on the School of Computer Science servers OR • Skilfully applies relevant scheduling concepts AND • Successfully implements an optimal solution that fully meets the assignment requirements with, at most, two minor errors or omissions AND • Demonstrates well-developed skills in producing human-readable source code that compiles on the School of Computer Science servers	8
• Competently applies relevant scheduling concepts AND • Implements an optimal or near-optimal solution that meets the basic coding requirements but with a few minor errors or omissions AND • Demonstrates well-developed skills in producing human-readable source code that compiles on the School of Computer Science servers OR • Skilfully applies relevant scheduling concepts AND • Successfully implements an optimal solution that fully meets the assignment requirements with, at most, two minor errors or omissions AND • Produces source code that that compiles on the School of Computer	7

Science servers but that is not always readily human-readable	
• Demonstrates basic skills in applying relevant scheduling concepts AND • Implements a sub-optimal solution to the problem, containing some errors or omissions but which meets the basic coding requirements AND • Demonstrates well-developed skills in producing human-readable source code that compiles on the School of Computer Science servers OR • Competently applies relevant scheduling concepts AND • Implements an optimal or near-optimal solution that meets the basic coding requirements but with a few minor errors or omissions AND • Produces source code that compiles on the School of Computer Science servers but that is not always readily human-readable	6
• Demonstrates basic skills in applying relevant scheduling concepts • Implements a sub-optimal solution to the problem, containing some errors or omissions but which meets the basic coding requirements • Produces source code that compiles on the School of Computer Science servers but that is not always readily human-readable	5
• Demonstrates limited skills in applying relevant scheduling concepts • Implements a flawed solution to the problem that fails to satisfy one or more basic coding requirements, contains one or more serious errors, contains one or more serious omissions, and/or contains multiple errors or omissions • Demonstrates basic programming skills, producing flawed source code that compiles on the School of Computer Science servers	4
• Demonstrates elementary skills in applying some relevant scheduling concepts • Implements a fatally flawed solution to the problem, containing major errors or omissions • Demonstrates sustained genuine engagement with the set programming problem but limited programming skills, producing fatally flawed source code that compiles on the School of Computer Science servers	3
• Demonstrates elementary understanding of some relevant scheduling concepts AND • Produces source code that compiles on the School of Computer Science servers and that contains some evidence of superficial engagement with the set programming problem OR • Demonstrates some skills in applying some relevant scheduling concepts AND • Produces relevant source code that does not compile on the School of Computer Science servers but that contains some evidence of sustained genuine engagement with the set programming problem	2

• Demonstrates minimal or no understanding of relevant scheduling concepts AND • Produces incomplete source code that compiles on the School of Computer Science servers and that contains minimal evidence of superficial engagement with the set programming problem OR • Demonstrates elementary understanding of some relevant scheduling concepts AND • Produces source code that does not compile on the School of Computer Science servers and that contains some evidence of superficial engagement with the set programming problem	1
• Disqualified by the COMP3520 examiner for any ONE of the following behaviours: ○ Engaging in an aggravated non-serious attempt in the source code ○ Engaging in a virtual non-attempt in the source code ○ Failing to submit source code OR • Disqualified by the Faculty of Engineering or the University due to academic dishonesty or misconduct in this assignment	0

Notes

1. If a *readme* file is not submitted, deduct one mark.
2. If a *readme* file is submitted but it does not represent serious work, deduct one mark.
3. If a *readme* file is submitted and it represents serious work but it is of unsatisfactory quality, deduct 0.5 marks.
4. If a working *makefile* is not submitted, deduct 0.5 marks.
5. Notwithstanding the first four points above, a negative mark shall not be awarded for the source code or the assignment; the minimum mark that may be awarded for the source code is zero.
6. Aggravating circumstances for non-serious attempts include the following (this is not an exhaustive list):
   a. Engagement in *Academic Dishonesty*;
   b. Without approval from the COMP3520 unit of study coordinator, recycling source code that had previously been submitted (whether by the same student or by another student) for assessment in any course at any University;
   c. Engagement in group work;
   d. Including frivolous or offensive material (this includes vulgar language, and inappropriate comments against examiners); or
   e. Prolonged failure to engage with the COMP3520 material without lawful excuse.
7. Virtual non-attempts include cases where a solution, or an attempt at a solution, for exercise 3 or 4 is submitted.