COMP3520 Operating Systems Internals Assignment 2 Marking Scheme

Code Functionality

Presentation of process data

Criteria	Marks
• Presents all of the process parameters in the correct format	2
• Presents some of the process parameters	1
• Performance does not meet the criteria above	0

Operation of HOST dispatcher

The following tests will be conducted:

- Operation of FCFS real-time scheduler – Staggered arrivals (fcfs.txt), simultaneous arrivals (fcfs1.txt) and gaps in arrivals (fcfs2.txt) cases;
- Operation of feedback scheduler in round robin mode – Staggered arrivals (rr.txt), simultaneous arrivals (rr1.txt) and gaps in arrivals cases (rr2.txt);
- Operation of three-level feedback scheduler – Staggered arrivals (feedback.txt) and feedback depth test (feedback1.txt) cases;
- Mixed dispatcher test – Combination of real-time and user processes (mixed.txt), and mixed feedback priorities (mixed1.txt) cases;
- Memory test – Standard memory allocation (memory.txt) and memory exceptions (memoryexcept.txt) cases;
- Resource test – Standard resource allocation (resource.txt) and resource exceptions (resourcexcept.txt) cases; and
- Comprehensive scheduling test (combined.txt).

For marking purposes, each file is categorised according to its importance in the test suite, whilst ensuring that the marks awarded for incomplete submissions is proportional to effort required and level of achievement.

The job files in the test suite and their assigned categories are as follows:

- fcfs.txt – category 3;
- fcfs1.txt – category 1;
- fcfs2.txt – category 1;
- rr.txt – category 5;
- rr1.txt – category 3;
- rr2.txt – category 3;
- feedback.txt – category 7;
- feedback1.txt – category 3;
- mixed.txt – category 4;
- mixed.txt1 – category 4;

- memory.txt – category 6;
- memoryexception.txt – category 2a;
- resource.txt – category 3;
- resourceexception.txt – category 2a; and
- combined.txt – category 2b.

In assessing process execution, synchronisation errors due to incorrect or missing waitpid() calls will be ignored, provided that they do not cause the dispatcher to freeze or crash. However, the correct use of waitpid() will be assessed separately.

The use of unapproved signals will result in penalties. Penalties must be applied under the "Penalties" section. However, in assessing process execution and final output, errors due to the use of an unapproved signal but which behaves in similar fashion to the signal prescribed in the assignment description will be ignored.

Category 1

Criteria	Marks
• Correct execution and final output	2
• Correct final output with one or more minor errors in execution	1
• Performance does not meet the criteria above	0

Category 2a

Criteria	Marks
• Correct execution and final output	3
• Rejects both bad jobs	2
• Rejects ONE of the two bad jobs	1
• Performance does not meet the criteria above	0

Category 2b

Criteria	Marks
• Correct execution and final output	3
• Correct final output with one or more minor errors in execution OR • Correct execution and final output but with incorrect or missing offsets	2
• Final output contains one error in process order	1
• Performance does not meet the criteria above	0

Category 3

Criteria	Marks
• Correct execution and final output	4
• Correct final output with one or more minor errors in execution	3
• Final output contains one error in process order	2
• Final output contains two errors in process order OR • Final output contains one error in process order AND one or more minor errors in execution that does not affect the final output	1
• Performance does not meet the criteria above	0

Category 4

Criteria	Marks
• Correct execution and final output	6
• Correct final output with one or more minor errors in execution	4
• Final output contains one minor error in process order	2
• Performance does not meet the criteria above	0

Category 5

Criteria	Marks
• Correct execution and final output	8
• Correct final output with one or more minor errors in execution	6
• Final output contains one error in process order	4
• Final output contains two errors in process order OR • Final output contains one error in process order AND one or more minor errors in execution that does not affect the final output	2
• Performance does not meet the criteria above	0

Category 6

Criteria	Marks
• Correct execution and final output including correct offsets	10
• Correct execution and final output but with one incorrect or missing offset	9
• Correct execution and final output but with two incorrect or missing offsets	8
• Correct execution and final output but with three incorrect or missing offsets	7
• Correct execution and final output but with four incorrect or missing offsets	6
• Correct execution and final output but with five incorrect or missing offsets	5
• Correct execution and final output but with six incorrect or missing offsets	4
• Correct execution and final output but with seven incorrect or missing offsets • Offsets are not phoney	3
• Correct execution and final output but with eight incorrect or missing offsets AND • Offsets are not phoney or identical OR • Correct execution and final output but with seven incorrect or missing offsets AND • Some offsets are phoney	2
• Correct execution and final output but all offsets are incorrect or missing AND • Offsets are not phoney or identical OR • Correct execution and final output but with eight incorrect or missing offsets AND • Offsets are not identical but some are phoney	1
• Performance does not meet the criteria above	0

Category 7

Criteria	Marks
• Correct execution and final output	12
• Correct final output with one or more minor errors in execution	9
• Final output contains one error in process order	6
• Final output contains two errors in process order OR • Final output contains one error in process order AND one or more minor errors in execution that does not affect the final output	3
• Performance does not meet the criteria above	0

Code Quality

Submission of files

Criteria	Marks
• Submits all required files with the correct names • No other files are submitted	1
• Performance does not meet the criteria above	0

Student name and number

Criteria	Marks
• Student no. and name are present on all source files, makefile and design document	2
• Student no. is present on all source files, makefile and design document OR • Student name is present on all source files, makefile and design document OR • Commensurate performance	1
• Performance does not meet the criteria above	0

Header files

Criteria	Marks
• Multiple header files complying with standard C programming conventions exist with the correct extension • Headers are arranged and structured logically	2
• Multiple header files complying with standard C programming conventions exist with the correct extension	1
• Performance does not meet the criteria above	0

Code commentary

Criteria	Marks
• Code is thoroughly and strategically commented • Demonstrates a thorough understanding of commenting practices	10
• Code is thoroughly commented • Demonstrates a thorough understanding of commenting practices	8 – 9
• Code contains substantial commentary that is written by the student • Demonstrates a sound understanding of commenting practices	6 – 7
• Code contains some commentary that is written by the student • Demonstrates a basic understanding of commenting practices	4 – 5
• Code contains little commentary that is written by the student • Demonstrates a limited understanding of commenting practices	2 – 3
• Code contains minimal or no commentary that is written by the student • Demonstrates minimal understanding of one or two commenting practices	1
• Performance does not meet the criteria above	0

Code structure

Criteria	Marks
• Writes exemplary code to implement a working host dispatcher with flair and precision • Demonstrates a thorough understanding of good code design practices	8
• Writes well-structured code to implement a working host dispatcher, possibly with some minor errors • Demonstrates a thorough understanding of good code design practices	6 − 7
• Writes code of satisfactory quality to implement a substantially or fully working host dispatcher • Demonstrates a sound understanding of good code design practices	4 − 5
• Writes code of variable quality to implement a substantially or fully working host dispatcher OR • Writes code of satisfactory quality to implement a single-level feedback dispatcher without memory allocation or resource allocation support AND • Demonstrates a basic understanding of good code design practices	2 − 3
• Writes code of generally poor quality to implement a substantially or fully working host dispatcher OR • Writes code of variable quality to implement a single-level feedback dispatcher without memory allocation or resource allocation support AND • Demonstrates a limited understanding of few good code design practices	1
• Performance does not meet the criteria above	0

NB: Any code that fails all tests listed under "Operation of HOST Dispatcher" will score zero marks for code structure. This will apply even if the process parameters have been printed correctly.

Dispatcher synchronisation

Criteria	Marks
• Correctly calls waitpid() whenever a process is suspended • Correctly calls waitpid() whenever a process is terminated	2
• ONE of the two points above fulfilled	1
• Performance does not meet the criteria above	0

Makefile header comment, layout and usability

Criteria	Marks
• Contains an appropriate header comment that briefly states the purpose of the makefile in the context of the assignment • Well laid out • Makefile works correctly	3
• Any TWO of the above points fulfilled	2
• Any ONE of the three points required to score full marks fulfilled	1
• Performance does not meet the criteria above	0

Bonus Code Marks

Award two bonus marks for ensuring that a running priority 1 process is not suspended if there are priority 3 processes waiting to be started or resumed but there are no other waiting processes (the priority must still be degraded).

Award four bonus marks for a correct implementation of the Buddy memory allocation scheme.

Maximum number of awarded bonus code marks is six.

Penalties

Deduct 3 marks for using a signal other than SIGINT to terminate a process.

Deduct 3 marks for using a signal other than SIGTSTP to suspend a process.

Deduct up to 15 marks for each intentional attempt at interfering with the marking routines.

Deduct up to 36 marks if the code needs editing in order to remove a compile error or segmentation fault error that prevents a proper assessment of the host dispatcher's performance from being made (-12 marks per error), provided that there are no more than three errors. If there are more than three errors, all marks for code functionality will be forfeited.

NB: Any marks gained by merely printing the sample output that has been supplied by the COMP3520 teaching staff will be forfeited.

Design Document

Bonus mark cap

The maximum number of bonus marks for this section is capped at 5 marks. The effect of this cap is that a student cannot score more than 50 marks in this section.

Description and discussion of memory allocation schemes (6 basic marks, no bonus marks)

Criteria	Marks
• Provides a thorough description of ALL prescribed memory allocation schemes • Provides an informed discussion of the advantages and disadvantages of these memory allocation schemes	6
• Provides a sound description of at least FOUR prescribed memory allocation schemes • Describes and/or explains the advantages and disadvantages of at least THREE of these memory allocation schemes	5
• Provides an adequate description of at least FOUR prescribed memory allocation schemes OR • Provides an adequate description of at least THREE prescribed memory allocation schemes AND • Outlines some advantages AND disadvantages of at least TWO of these memory allocation schemes OR • Provides an adequate description of TWO prescribed memory allocation schemes AND • Provides a limited description of the remaining prescribed memory allocation schemes AND • Outlines some advantages and/or disadvantages of at least THREE of these memory allocation schemes	4
• Provides an adequate description of THREE prescribed memory allocation schemes OR • Provides an adequate description of TWO prescribed memory allocation schemes AND • Outlines some advantages and/or disadvantages of at least ONE of these memory allocation scheme OR	3

• Provides an adequate description of TWO prescribed memory allocation schemes AND • Provides a limited description of at least TWO other prescribed memory allocation schemes OR • Provides a limited description of ALL prescribed memory allocation schemes AND • Outlines some advantages and/or disadvantages of at least FOUR of these memory allocation schemes	
• Provides an adequate description of TWO prescribed memory allocation schemes OR • Provides a limited description of at least FOUR prescribed memory allocation schemes OR • Provides a limited description of at least THREE prescribed memory allocation schemes AND • Outlines some advantages and/or disadvantages of at least TWO of these memory allocation schemes	2
• Provides an adequate description of ONE prescribed memory allocation scheme OR • Provides a limited description of at least TWO prescribed memory allocation schemes OR • Repeats relevant material verbatim or near-verbatim from another cited or referenced source	1
• Performance does not meet the criteria above	0

Justification of choice of memory allocation scheme (2 basic marks, no bonus marks)

Criteria	Marks
• Identifies the memory allocation algorithm used in the code and clearly justifies this choice in the context of the assignment	2
• Identifies the memory allocation scheme used in the code and makes a serious attempt at justifying this choice	1
• Performance does not meet the criteria above	0

Description and discussion of the structures used by the dispatcher (5 basic marks + 1 bonus mark)

Criteria	Marks
• Provides a thorough description of ALL dispatcher structures that are relevant to the host dispatcher implementation • Provides an informed discussion of the roles of the dispatcher structures and their suitability	6
• Provides a sound description of at least FOUR dispatcher structures that are relevant to the host dispatcher implementation • Describes and/or explains the roles of these dispatcher structures in the host dispatcher and their suitability	5
• Provides an adequate description of at least FOUR dispatcher structures OR • Provides an adequate description of THREE dispatcher structures AND • Outlines the roles of at least TWO dispatcher structures in the host dispatcher and their suitability OR • Provides an adequate description of TWO dispatcher structures AND • Provides a limited description of at least TWO other dispatcher structures AND • Outlines the roles of at least THREE dispatcher structures in the host dispatcher and their suitability	4
• Provides an adequate description of THREE dispatcher structures OR • Provides an adequate description of TWO dispatcher structures AND • Outlines the roles of at least ONE dispatcher structure in the host dispatcher and their suitability OR • Provides an adequate description of TWO dispatcher structures AND • Provides a limited description of at least TWO other dispatcher structures OR • Provides a limited description of at least FOUR dispatcher structures AND • Outlines the roles of at least THREE dispatcher structures in the host dispatcher and their suitability	3

• Provides an adequate description of at least TWO dispatcher structures OR • Provides a limited description of at least FOUR dispatcher structures OR • Provides a limited description of at least THREE dispatcher structures AND • Outlines the roles of at least TWO dispatcher structures in the host dispatcher and their suitability	2
• Provides an adequate description of ONE dispatcher structure OR • Provides a limited description of at least TWO dispatcher structures OR • Repeats relevant material verbatim or near-verbatim from another cited or referenced source	1
• Performance does not meet the criteria above	0

Description and justification of the program structure and individual modules (12 basic marks + 3 bonus marks)

Criteria	Marks
• Provides a comprehensive description of the overall program structure, its modules and major functions • Provides detailed and clear justification of the program structure in the context of the assignment	15
• Provides a thorough description of the overall program structure, its modules and major functions • Provides clear or detailed justification of the program structure in the context of the assignment	13 – 14
• Provides a sound description of the overall program structure and most of, or all of, its modules and major functions • Provides some justification of the program structure in the context of the assignment	10 – 12
• Provides an adequate description of the overall program structure and some of its modules and/or major functions • Makes a serious attempt at justifying the program structure in the context of the assignment	7 – 9
• Provides a limited description of the overall program structure and at least TWO modules and/or major functions	4 – 6
• Provides a limited description of the overall program structure OR • Provides a limited description of at least ONE module or major function OR • Repeats relevant material verbatim or near-verbatim from another cited or referenced source	1 – 3
• Performance does not meet the criteria above	0

Discussion of dispatching scheme, shortcomings and possible improvements (20 basic marks + 5 bonus marks)

Criteria	Marks
• Demonstrates extensive knowledge and understanding of operating systems concepts relevant to the question • Provides an insightful discussion of the reasons why multilevel dispatching schemes are important for process management • Provides an insightful comparison of the HOSTD dispatcher and other multilevel dispatching schemes used in at least TWO different real operating systems • Outlines a range of shortcomings of the HOST dispatcher and possible improvements to address them with clear detailed justification • Refers to well-chosen examples to inform response • Presents a sustained, cohesive and logical response using appropriate information and ideas from a range of sources	25
• Demonstrates thorough knowledge and understanding of operating systems concepts relevant to the question • Provides an informed discussion of the reasons why multilevel dispatching schemes are important for process management • Provides a detailed comparison of the HOSTD dispatcher and other multilevel dispatching schemes used in at least TWO different real operating systems • Outlines a range of shortcomings of the HOST dispatcher and possible improvements to address them with justification • Refers to relevant examples to inform response • Presents a cohesive and logical response using appropriate information and ideas from a range of sources	21 – 24
• Demonstrates sound knowledge and understanding of operating systems concepts relevant to the question • Explains why multilevel dispatching schemes are important for process management • Provides a sound comparison of the HOSTD dispatcher and at least ONE multilevel dispatching scheme used in a real operating system • Outlines a range of shortcomings of the HOST dispatcher and possible improvements to address them • Refers to relevant examples where appropriate • Presents a structured extended response using appropriate information and ideas from some sources	16 – 20
• Demonstrates basic knowledge and understanding of a representative range of operating systems concepts relevant to the question • Outlines at least ONE reason why multilevel dispatching schemes are important for process management • Provides a superficial comparison of the HOSTD dispatcher and at least ONE multilevel dispatching scheme used in a real operating system • Outlines at least ONE shortcoming of the HOSTD dispatcher and at least ONE possible improvement to the HOSTD dispatcher	11 – 15

• Refers to at least ONE relevant example to illustrate an operating systems internals concept • Presents an extended response using some information relevant to the study of operating systems internals	15
• Demonstrates limited knowledge and understanding of some operating systems concepts relevant to the question • Attempts to engage with some aspects of the question • Uses limited information relevant to the study of operating systems internals	6 – 10
• Demonstrates very limited knowledge and understanding of few operating systems concepts relevant to the question AND • Uses very limited information relevant to the study of operating systems internals OR • Repeats relevant material verbatim or near-verbatim from another cited or referenced source	1 – 5
• Performance does not meet the criteria above	0

Acknowledgements