

CAB403 Assignment Marking Criteria

Student Name(s): Levinard

Student Number(s): n9440160

Total Marks: 95/100

NOTE: Programs that do not compile on the Linux command line will receive a mark of zero (0). Submission must be implemented in the C programming language using BSD sockets on the Linux operating system which has been used in the weekly practicals.

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

Task 1: (48/50)

| Criteria | Marks |
|--|-------|
| Server command line parameter – configurable port & default port | 3/3 |
| Server authenticates client using data in "Authentication.txt" file | 6/6 |
| Server loads & tokenises "hangman_text.txt" file correctly | 6/6 |
| Server randomly selects a two-word phrase | 3/3 |
| Server updates Leader Board when a client finishes playing a Hangman game | 5/5 |
| Server exits gracefully upon receiving SIGNAL (ctrl + c) | 2/3 |
| Client command line parameters | 3/3 |
| Client menu implementation | 3/3 |
| Client plays Hangman as per assignment specifications when user selects "Play Hangman" | 6/6 |
| Client displays leader board as per assignment specifications when user selects option "Show Leader Board" | 6/6 |
| Client exits gracefully when user selects "Quit" option from menu | 3/3 |
| Description of the data structure that is used for the Leader Board in your report | 2/3 |

Task 2: (19/20)

| Criteria | Marks |
|---|-------|
| Multithreaded implementation | 10/ |
| Process synchronization | 6/6 |
| Description of how the critical-section problem is handled in your report | 3/4 |

Task 3: (14/16)

| Criteria | Marks |
|--|-------|
| Thread pool creation | 7/7 |
| Thread pool use | 3/3 |
| Thread pool cleaning | 2/3 |
| Description of how the thread pool is created and managed in your report | 2/3 |

PROGRAM QUALITY

Marks: (10/10)

| Criteria | Marks |
|---|-------|
| Program structure & readability | 2/2 |
| Program performance | 2/2 |
| Resource management | 3/3 |
| Program reliability (e.g. run time errors, deadlocks, file I/O) | 3/3 |

REPORT QUALITY

Marks: (4/4)

| Criteria | Marks |
|---|-------|
| Statement of completeness | 2/2 |
| Instructions on how to compile and run your program | 2/2 |

Comments:

Very Good. Overall You have done a good job.

NOTE: Allocation of marks depends on which tasks are attempted.

| Task | Functional and Non-functional Requirements | Code Quality | Report | Maximum Marks |
|---------------------------|--|--------------|--------|---------------|
| Task 1 only | 50 | 10 | 4 | 64 |
| Task 1 and Task 2 | 70 | 10 | 4 | 84 |
| Task 1, Task2, and Task 3 | 86 | 10 | 4 | 100 |