CAB403 Systems Programming Semester 2, 2018

CAB403 Assignment Marking Criteria

Student Name(s): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student Number(s): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Total Marks: \_\_\_\_\_ /**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.**

**Programs which crash (or “segmentation fault”) during testing of a criteria will receive zero (0) for the criteria, and any which depend on it. For example, a program which crashes during login will not be tested for gameplay.**

**If you attempt Task 2 and 3, only one (1) server program is required to be submitted, incorporating all the functionality from the previous tasks.**

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

Task 1: (\_\_\_\_/50)

|  |  |
| --- | --- |
| **Criteria** | **Marks** |
| Network byte order is used for transmitting multi-byte data types (server and client) | /2 |
| Server command line parameter – configurable port & default port | /2 |
| Server authenticates client using data in “*Authentication.txt”* file | /5 |
| Server exits gracefully upon receiving SIGNAL (ctrl + c) | /3 |
| Client command line parameters | /2 |
| Client menu implementation | /3 |
| Client exits gracefully when user selects “Quit” option from menu | /2 |
| Quitting part-way through a game resets the playfield | /2 |
| Random number generator is seeded as per assignment specification | /1 |
| Mines are placed on playfield using algorithm from assignment specification | /1 |
| Revealing a tile with adjacent mines shows the number of adjacent mines | /2 |
| Revealing a tile with zero adjacent mines reveals neighbouring zero mines | /4 |
| Revealing a tile with a mine ends the game, and displays the full playfield | /2 |
| Placing a flag on a tile with a mine decrements the remaining mine count | /2 |
| Placing a flag on a tile without any mine has no effect on remaining count | /2 |
| Placing flags on all mines results in a game win, and the full playfield displayed | /4 |
| Time taken to complete the game is measured and displayed correctly | /2 |
| Leader board is updated after winning a game | /3 |
| Leader board is displayed as per assignment specifications | /3 |
| Description of the data structure that is used for the Leader Board in your report | /3 |

1 | P a g e

CAB403 Systems Programming Semester 2, 2018

Task 2: (\_\_\_\_/20)

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

Task 3: (\_\_\_\_/16)

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

PROGRAM QUALITY

Marks: (\_\_\_\_/10)

|  |  |
| --- | --- |
| **Criteria** | **Marks** |
| Program structure & readability | /[[1]](#footnote-1) |
| Program performance | /2 |
| Resource management | /3 |
| Program reliability (e.g. run time errors, deadlocks, file I/O) | /3 |

REPORT QUALITY

Marks: (\_\_\_\_/4)

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

***Comments:***

***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** |

1. | P a g e

   [↑](#footnote-ref-1)