Project 4 – Synchronization – Grading Sheet/Rubric

Evaluation?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Grader: |  |  | Student Name: |  |  |  |
| Date/Time: |  |  | Student Name: |  |  |  |
| Team ID: |  |  | Student Name: |  |  |  |
| Late?: |  |  |  |  |  |  |
| Checkpoint?: |  |  |  | Project Score: |  | / 120 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Earned** | **Weight** | **Task ID** | **Description** |
| \_\_\_\_\_ | 0 | 0 | Primitives – Students must use a different primitive for Part 1 and Part 2. If the same primitive is used for both, students may choose which part will be graded. The other part is to be assigned a score of 0. |
| \_\_\_\_\_ | 20 | 1 | Part 1 – Threads/processes implemented correctly with a high degree of parallelism. Prerequisite: Task 0. |
| \_\_\_\_\_ | 20 | 2 | Part 1 – Correct mutual exclusion and prevention of deadlocks. Prerequisite: Task 1. |
| \_\_\_\_\_ | 20 | 3 | Part 1 – Solution ensures fairness/starvation prevention and good explanation in problem1\_explanation.txt. Prerequisite: Task 1. |
| \_\_\_\_\_ | 20 | 4 | Part 2 – Threads/processes implemented correctly with a high degree of parallelism. Prerequisite: Task 0. |
| \_\_\_\_\_ | 20 | 5 | Part 2 – Correct mutual exclusion and prevention of deadlocks. Prerequisite: Task 4. |
| \_\_\_\_\_ | 20 | 6 | Part 2 – Solution ensures fairness/starvation prevention and good explanation in problem2\_explanation.txt. Prerequisite: Task 5. |

Grader Notes: