

Group 01

| | Rubrics | Full Points | Points |
|--|--|--------------------|---------------|
| | | | |
| | 1. Reservation management | 20 | |
| | No compile error (Kernel / Modules) | 2 | 2 |
| Correctly reserve a new task (set_rsv) | If pid is 0, apply reservation for calling task correctly | 3 | 3 |
| | If pid is a specific pid, apply reservation for that task correctly | 3 | 3 |
| | Return -1 if a task has already a reservation | 2 | 2 |
| | Run by specified period T and C | 5 | 5 |
| Correctly cancel a new task (cancel_rsv) | Correctly cancel the specified pid task from the reservation framework | 3 | 0 |
| | No kernel panic or errors after cancellation | 2 | 0 |
| | | | |
| | 2. RT priority assignment | 20 | |
| Correctly assign RT priorities for multiple tasks based on their periods | No compile error | 5 | 5 |
| | Multiple tasks can be reserved (at least three) | 5 | 5 |
| | Correctly RT priority assignments | 5 | 0 |
| | Correctly schedule RT scheduling (SCHED_FIFO) for reservation tasks | 5 | 0 |
| | | | |
| | 3. Periodic task support | 20 | |
| | Correctly make a reserved task sleep mode | 15 | 15 |
| | Return correct values based on reservation (0 or -1) | 5 | 5 |
| | | | |
| | 4. Computation time tracking | 30 | |
| Implementation of accumulator | Keep track of computation time across context switches | 10 | 5 |
| | Reset the accumulator at every period | 10 | 10 |
| | Print message when a task has exceeded its budget | 10 | 10 |
| | | | |
| | 5. Budget overrun notification | 5 | |
| | Implemented and working correctly | 5 | 0 |
| | | | |
| | 6. Write up | 10 | 10 |
| | | | |
| | Total | | 80 |