

Assignment 6

Deadline: Friday **05.01.2024** 23:55

Notes:

- Solve the assignment **on your own** — no groups allowed.
- Hand-written solutions will not be accepted, except for graphs and diagrams.
- If you hand in non-pdf files or multiple files, name your submission as `stla23_06_SURNAME.zip`, replacing `SURNAME` with your surname. Otherwise `stla23_06_SURNAME.pdf`. Also include your full name in the submitted PDF.
- Submit your solution via **Ilias**.

```
1 while true do
2   noncritical section
3   P(s)
4   critical section
5   V(s)
```

Algorithm 1: Mutual exclusion algorithm for a single process using a shared semaphore s

Exercise 6.1 Mutual exclusion using a semaphore

20 Points

Consider the mutual exclusion algorithm in Algorithm 1.

- Write a TLA⁺ specification of this algorithm for N processes. Include the attached module `Semaphore` with `EXTENDS`. **(10 points)**
- Specify a invariant that ensures mutual exclusion. Prove the invariant for your specification with TLC. **(3 points)**
- Specify a liveness property that ensures that every process that calls $P(s)$ eventually enters the critical section. Find the weakest fairness condition that is necessary to prove this property for this algorithm. Prove that the property holds for your specification under this fairness condition. **(6 points)**

Exercise 6.2 Your own Mutex (10 Bonus Points)

0 Points

Specify a mutual exclusion algorithm for N processes of your own choice/design in TLA⁺ which is not the one in Algorithm 1. The algorithm should ensure mutual exclusion and that every process that waits for the critical section eventually enters it under a weak fairness condition. Prove the correctness of your algorithm with TLC. **(10 bonus points)**

Total: 20 Points