Selina Brinnich, Alexander Dietrich

Exercise 1: Dining Philosophers

1. What are the necessary conditions for deadlocks (discussed in the lecture)?

- Mutual exclusion: multiple threads need to use the same resource concurrently
- Hold and wait: a thread can request and wait for a resources while currently holding another one
- No preemption: resources are only released by the thread currently blocking it and not by stealing
- Circular wait: multiple threads are waiting for each other in a circular pattern

2. Why does the initial solution lead to a deadlock (by looking at the deadlock conditions)?

- Mutual exclusion: each fork is used by two philosophers concurrently
- Hold and wait: each philosopher requests the second fork while currently holding the first fork
- No preemption: forks can not be stolen by other philosophers if they are currently held by a philosopher
- Circular wait: each philosopher takes the left spoon first -> when all of them do this simultaneously they create a circular wait on the right spoon afterwards

3. Does this strategy resolve the deadlock and why?

- Yes it does, because it breaks the circular wait condition. Philosophers sitting next to each other always take the fork between them either both as first one (which will leave the second philosopher trying to pick it up waiting already for the first spoon), or both as second one (which will lead the first philosopher to finish eating and giving his forks back so the second one can eat as well).

4. Measure the time spent in waiting for fork and compare it to the total runtime.

- The values are almost the same over different runthroughs. The percentage of waiting time of the total execution time is about 15%, although this depends (mainly) on the number of philosophers. The more Philosophers the higher the total waiting time compared to the execution time.

- E.g. execution with parameters 20, 20, 20:

The highest waiting time for a locked fork: 84 ms
The average waiting time for a fork: 7 ms
The total waiting time for all threads: 14051 ms
The total execution time for all threads: 98798 ms
The percentage of waiting time of the total execution time over all threads: 0.142219 (14.2%)

- E.g. execution with parameters 100, 20, 20:

The highest waiting time for a locked fork: 107 ms
The average waiting time for a fork: 11 ms
The total waiting time for all threads: 112503 ms
The total execution time for all threads: 717402 ms
The percentage of waiting time of the total execution time over all threads: 0.15682 (15.7%)

---

5. Can you think of other techniques for deadlock prevention?

- E.g. breaking the Hold and wait condition: putting the first fork down and think again when second fork is currently not available