Laboratory for Parallel Programming
Prof. Dr. Felix Wolf

TECHNISCHE UNIVERSITÄT DARMSTADT

Marcus Ritter, marcus.ritter@tu-darmstadt.de

Advanced Multithreading in C++ (WS 21/22)

Exercise 2

Please solve the following tasks by November 30th, 2021. The results are not graded, but a solution is discussed on November 30th, 2021.

Task 1

A programmer wants to use asynchronous tasks to make her code run more concurrently. Therefore, she wrote a main thread, which reads the content of a file, and a worker thread to process the read data. In order to work, the main thread needs to notify the worker threads to start processing a part of the data once it has been read. Listing 1 shows her code skeleton. To do the necessary synchronization she wants to use conditional variables.

Listing 1: Sample program

```
1 void processData(){
2     // do the processing
3 }
4
5 void readFile(){
6     // read the input file
7 }
```

1. Write a code that uses conditional variables to synchronize the main with the worker thread.
2. What are the disadvantages of using conditional variables for synchronization?
3. What other approaches exist to solve this problem? Write a code example for each of them.
4. The programmer wants to extend the code so that multiple tasks can respond to the main thread. Write an example code for this scenario using the promis/future approach and explain the changes necessary to notify multiple reacting tasks.
5. C++11 introduced a template for atomic types `std::atomic<T>`. Solve the problem of task 4 by the following: All threads are busy-waiting on an atomic int, and the master sets the value to the number of threads that should wake up.