

TDT4200-PS7

Optional Assignment - Parallel I/O

Jacob O. Tørring and Zawadi Svela

Deadline: Wed November 17, 2021 at 22:00 on BlackBoard
Evaluation: Pass/Fail

- **This assignment must be done individually and w.o. help from anyone but The TDT4200 staff.** We encourage that you post clarification questions on the Forum in BB, so all can benefit. All sources used found on the internet or otherwise must be referenced
- Code should not use other external dependencies aside from the ones we specified. Do not deliver any other files than those specified in the Deliverables section.

Parallel I/O and PThreads

In this exercise, you will parallelize writing images to files as part of the pre-written serial program for Beier-Neely from PS6.

Important: When you deliver your code, make sure that you only deliver the files explicitly asked for. This will only include code you have edited yourself, and not binary files, example input, etc. A consequence of this is that *you are not allowed to utilize external libraries not already included in the hand-out code.*

Setup

For this problem set we recommend using the PCs at the Cybele Lab or the Snotra compute cluster as described under Course Information.

Before you begin the tasks, you should start by testing the program. Compile the serial implementation using Make and run the program.

```
./morph images/man9.jpg images/man10.jpg lines/lines-man9-man10.txt output-png/ 10
```

This should create a series of images in the output-folder, each morphed some way between the source and destination images.

Tasks

The following tasks are all marked with corresponding TODO comments in the handout code. After each task, you should ensure that the program compiles, though it might not run after some of the changes.

- A Setup thread handlers.
- B Extract the code lines that you want to parallelize into a separate function.
- C Run the function with each pthread and respective struct.
- D Join the threads.
- E Measure the speedup.

Deliverables

Deliver the finished morph.cu file on Blackboard. It should run error-free and produce the correct output. **Do not zip the file.**