



# Task Scheduler

Parallel Computing

## Goals

- ★ Schedule a graph of tasks in parallel.
- ★ **Relevant videos:** if you want to get started as quick as possible, follow the videos annotated with “fast track”. Of course, all videos should be watched eventually.
  - Why shared read/write memory?
  - Atomic
  - Compare-and-swap
  - Task parallelism (**fast track**)
  - Dynamic task scheduling (**fast track**)
  - Task Parallelism with OpenMP
  - Producer-consumer problem (**fast track**)
  - Dining philosophers problem (**fast track**)

## Deliverables

1. Starting code: [https://github.com/ptal/task\\_scheduler](https://github.com/ptal/task_scheduler)
2. **Report:** Write a brief report describing your system and answering some questions in the README.

## Rules

1. You can discuss your design and your results on Discord or orally, but please don’t share your code.
2. This is a solo project.

## Exercise 1 – Task Scheduler

The description of this assignment is available in the README of the repository.