

# Lab Worksheet 02 - Scheduling

## Objectives

1. Assimilate the concepts associated with process scheduling.
2. Learn to assess the consequences of process scheduling policies.

## Exercise 1: Toying around

Start by [downloading this lab's scheduling\\_policy\\_simulator](https://newclasses.nyu.edu/access/content/group/51ce8755-5381-4dd5-bae5-8ae3b3c862d0/Worksheets/Skeleton-Code/labOnScheduling.tgz) (<https://newclasses.nyu.edu/access/content/group/51ce8755-5381-4dd5-bae5-8ae3b3c862d0/Worksheets/Skeleton-Code/labOnScheduling.tgz>), which only implements one scheduling policy (FCFS).

The simulator parses the 'tasks' text file provided in the archive: this file describes the list and characteristics of tasks to be scheduled. The file is organized as follows: the first column gives the name of the task, the second one the number of computation cycles the task must execute, and the last value represents the insertion date of the task in the system. Look at the content of the tasks file:

```
$ cat tasks
T1 10 0
T2 8 2
T3 5 26
T4 7 2
```

You are now ready to execute the scheduling policies simulator:

```
$ make build
$ make run
```

You should see a trace explaining how the tasks were sequenced by the scheduling simulator. Check that the trace is correct.

Now try a different set of tasks.

Create a "tasks2" file with the following task parameters:

```
T1 10 10
T2 12 2
T3 5 3
```

And run the simulator with

```
$ bin/sched-simulator "tasks2"
```

## Exercise 2: Enter the metrics

Modify the simulator so that every simulation run ends with a display of the following statistics:

- The turnaround time for each task
- The penalty rate for each task
- The average waiting time

## Exercise 3: You're the captain now

Implement the Shortest Job First and the Shortest Remaining Time First policies.  
Draw a table to compare the statistics of all three policies.