Math 140: Tuesday, Sep 24, Day 9 Notes

Agenda for Tuesday, Sep 24,2024	
Chapter 2 Scheduling	**1**

Coming up

- Office Hour **today 1-2:15pm** in C162

Scheduling

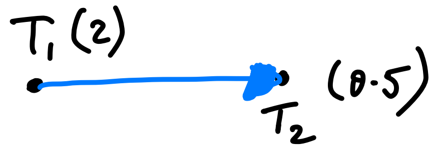
The problem of scheduling is fairly universal. Contractors need to schedule workers and subcontractors to build a house as quickly as possible. A magazine needs to schedule writers, editors, photographers, typesetters, and others so that an issue can be completed on time.

Example 1. An auto shop is converting a car from gas to electric.

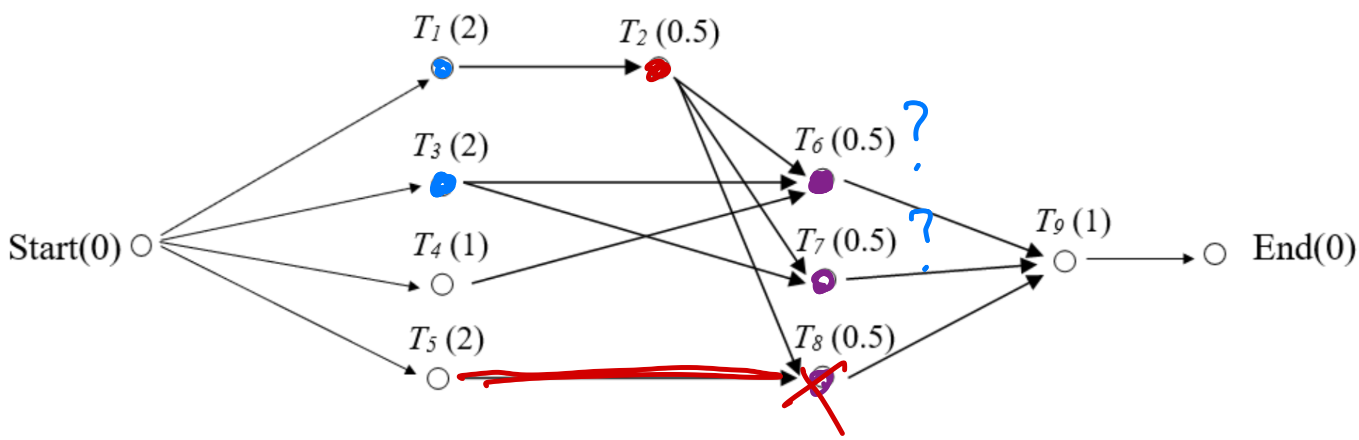
- Task 1: Remove engine and gas parts (2 days)
- Task 2: Steam clean the inside of the car (0.5 day)
- Task 3: Buy an electric motor and speed controller (2 days for travel)
- Task 4: Construct the part that connects the motor to the car's transmission (1 day)
- Task 5: Construct battery racks (2 days)
- Task 6: Install the motor (0.5 day)
- Task 7: Install the speed controller (0.5 day)
- Task 8: Install the battery racks (0.5 day)
- Task 9: Wire the electricity (1 day)

Digraph is a directed graph in which the vertices represent tasks and the edges are arrows used to show ordering.

Example 2.



A complete digraph for our car conversion would look like this:



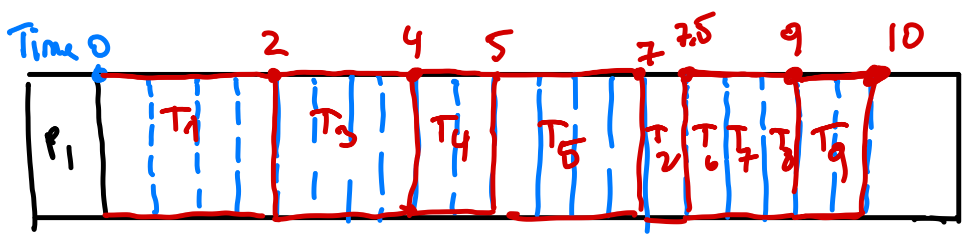
Processors are workers completing the task

ex: Auto shop
processors are the mechanics.

Assumptions:

- Every processor can do every task
- Only one processor can work at a task at a time

One processor Solution:



Finishing time is the time it takes to complete all tasks. It depends on the number of processors.
with 1 processor it take 10 days to complete this project.

Optimal schedule is the schedule with the shortest possible finishing time.

Critical time is the absolute minimum time to complete the job, regardless of the number of processors working on the tasks.

List Processing Algorithm

To create a more procedural approach, we might begin by somehow creating a priority list.
Priority List A priority list is a list of tasks given in the order in which we desire them to be completed.
Priority lists can be given to you or you have to create them.

Example 3. Schedule the project with two processors, using the priority list:

$$T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$$

Solution for our example when we are given the priority list.

Algorithm:
Time 0: Mark ready tasks by circling them.

$$T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$$

Assign tasks to processor and mark those tasks as in progress.

$$\cancel{T_1}, \cancel{T_3}, T_4, T_5, T_6, T_7, T_8, T_2, T_9$$

Write the schedule up to here.

Time:	0	1	2	3	4	5	6	7	8	9	10
P_1	T_1										
P_2	T_3										

We jump to the time when the next task completes, which is at time 2.

Time 2: Both processors complete their tasks. We mark those tasks as complete (by crossing them). With Task 1 complete, Task 2 becomes ready:

T_2 became available once T_1 and T_3 were done

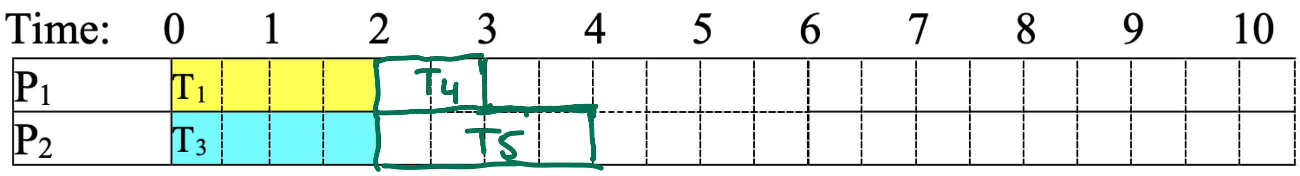
Priority list status:

$$\cancel{T_1}, \cancel{T_3}, T_4, T_5, T_6, T_7, T_8, T_2, T_9$$

We assign the next ready tasks on the list

$$\cancel{T_1}, \cancel{T_3}, \underset{P_1}{\cancel{T_4}}, \underset{P_2}{\cancel{T_5}}, T_6, T_7, T_8, T_2, T_9$$

Write the schedule up to here.



Time 3: Processor 1 has completed T_4. Completing T_4 does not make any other tasks ready (note that all the rest require that T_2 be completed first).

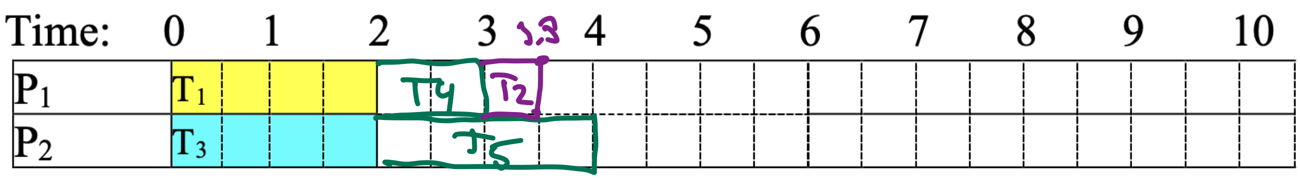
Priority list status:

$$\cancel{T_1}, \cancel{T_3}, \cancel{T_4}, \cancel{T_5}, T_6, T_7, T_8, T_2, T_9$$

We assign the next ready tasks on the list

$$\cancel{T_1}, \cancel{T_3}, \cancel{T_4}, \cancel{T_5}, T_6, T_7, T_8, \cancel{T_2}, T_9$$

Write the schedule up to here.



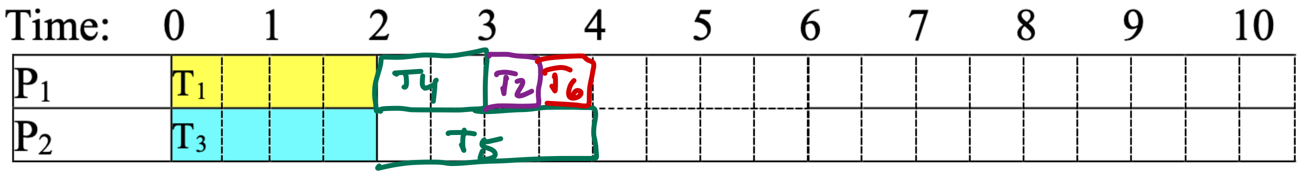
Time 3.5: Processor 1 has completed T_2. Completing T_2 causes T_6 and T_7 to become ready. We assign T_6 to P_1

Priority list status:

$$\cancel{T_1}, \cancel{T_3}, \cancel{T_4}, \cancel{T_5}, T_6, T_7, T_8, \cancel{T_2}, T_9$$

P_1 gets T_6

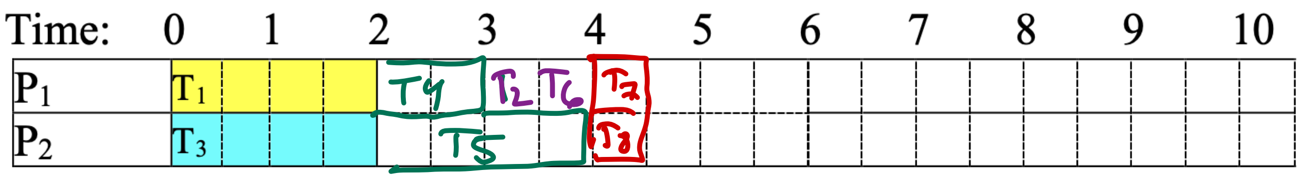
Write the schedule up to here.



Time 4: Both processors complete their tasks. The completion of T_5 allows T8 to become ready. We assign T_7 to P_1 and T_8 to P_2.
Priority list status:

$$\cancel{T_1}, \cancel{T_3}, \cancel{T_4}, \cancel{T_5}, \cancel{T_6}, \cancel{T_7}, \cancel{T_8}, \cancel{T_2}, T_9$$

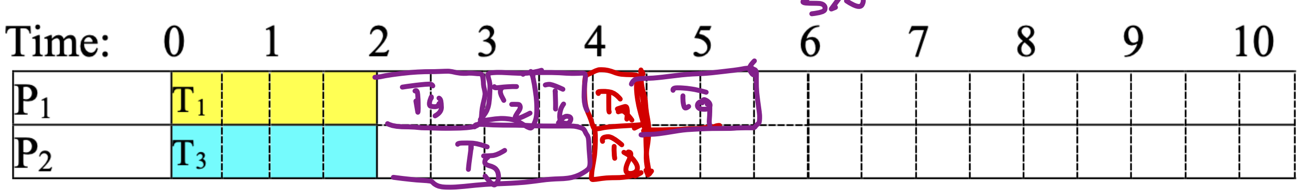
Write the schedule up to here.



Time 4.5: Both processors complete their tasks. T_9 becomes ready, and is assigned to P_1. There is no ready task for P_2 to work on, so P_2 **idles**.
Priority list status:

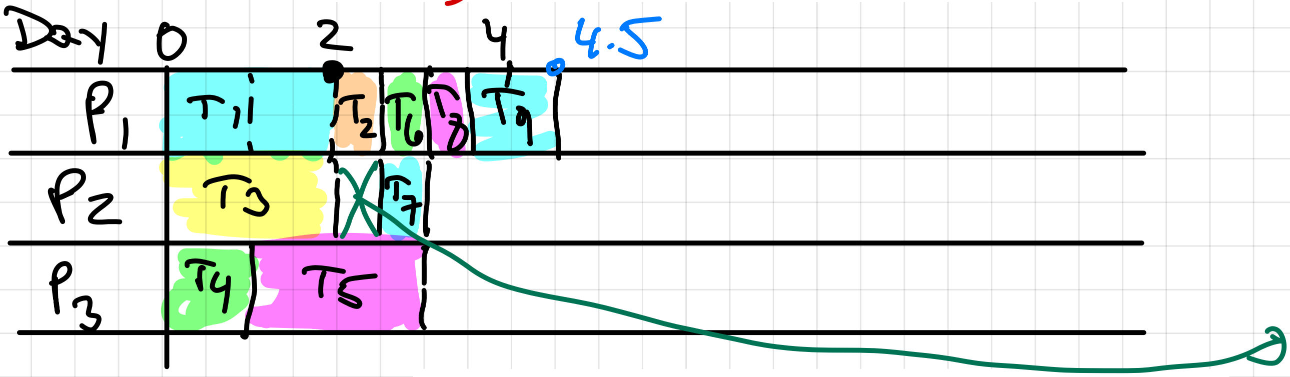
$$\cancel{T_1}, \cancel{T_3}, \cancel{T_4}, \cancel{T_5}, \cancel{T_6}, \cancel{T_7}, \cancel{T_8}, \cancel{T_2}, T_9$$

5.5 days completion time



With the last task completed, we have a completed schedule, with finishing time 5.5 days

$T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$



idle time (can't work on any tasks)

Day 0 : $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

Day 1 : $P_3 \to T_5$ $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

Day 2 : $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

Day 2.5 : $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

Day 3 : $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

Day 3.5 : $T_1, T_3, T_4, T_5, T_6, T_7, T_8, T_2, T_9$

completion time 4.5 days