06.12.21 ALS TD 11 — Algos d'approximation — un seul exo parce que c'était le jour du CC3.

Exercice 1

1)
```
∀ machine m
    charge[m] = 0;
    taches[m] = ∅;
∀ tâche j :
    m = minMachineCharge
    charge[m] += t_j ;
    tâches[m].add(j);
    update(m);
```
m tours de boucle

n tours de boucle

minMachineCharge : O(1)

update : O(log m)

with a binary heap

$\Rightarrow O(n \log(m))$

2) L'algo donne $m-1$ machines avec charge $m-1$
et 1 machine avec charge $m-1+m = 2m-1$

Solution optimale

$m-1$ machines avec m tâches à 1 sec chacune $\Rightarrow$ charge m

et 1 machine avec la tâche à m secondes $\Rightarrow$ charge m

Facteur d'approx

$\dfrac{\text{notre algo}}{\text{optimal}} = \dfrac{2m-1}{m} \approx 2$

3) Dans un scénario parfait, on peut répartir les tâches de façon à ce que toutes les machines aient la même charge.

Donc $T_{parfait} = \dfrac{1}{m} \sum_j t_j$

Mais en vrai, il se peut qu'il y ait des tâches plus longues que d'autres : ex pour 2 machines et 2 tâches de longueur 1 et 4.

Donc la valeur optimale sera plus grande que la valeur parfaite.

Donc $T^* \geq \dfrac{1}{m} \sum_j t_j$

4) Chaque tâche n'est prise que par une seule machine : on ne peut pas diviser son temps.

Donc $T^* \geq \max_j t_j$ car une machine doit s'occuper de la tâche de longueur max, donc le pire temps machine est au ⊖ égal au pire temps des tâches.

5) Avant d'ajouter la tâche j, M_i était la machine avec la plus petite charge parmi les m machines.

En particulier, sa charge était inférieure ou égale à la moyenne des charges.

Donc $T_i - t_j \leq T_i \leq \dfrac{1}{m} \sum_{k=1}^{m} T_k$