École polytechnique — Majeure informatique
Promotion 2004

Cours « Conception et analyse d'algorithmes »
Devoir à rédiger n° 1 – Programmation dynamique
à rendre le 31 janvier 2007 en début de pc

Ordonnancement pour minimiser le nombre de tâches en retard

On considère de nouveau un problème d'ordonnancement de tâches, en supposant cette fois que les tâches T_i $(1 \leqslant i \leqslant n)$ ont une durée d_i. Pour chaque tâche sont fixés une date d'échéance e_i et une pénalité p_i à acquitter quand la tâche n'est pas terminée avant la date e_i. On veut trouver un ordonnancement des tâches sur une machine unique qui minimise la somme des pénalités de retard.

i. Montrer que lorsqu'on connaît l'ensemble des tâches qui sont exécutées sans retard, il est facile de trouver un ordonnancement optimal.

ii. Montrer qu'on peut déterminer l'ensemble des tâches à exécuter sans retard par programmation dynamique. Les tâches étant classées selon leur date d'échéance, on considérera pour cela la durée minimale d'un ordonnancement sans retard d'un sous-ensemble de tâches inclus dans $\{T_1, \cdots, T_j\}$ et de pénalité totale supérieure ou égale à p.