# Algorithmique et structure de données
# Filière Electronique
# Corrigé du TD 6

## Un algorithme glouton pour la mise en boîte

On a $n$ objets de poids, $p_0, p_1, \ldots, p_{n-1}$ que l'on veut placer dans un nombre minimum de boîtes sachant que chaque boîte peut contenir au plus un poids *pmax* donné.

L'algorithme glouton consiste à considérer les objets chacun à leur tour dans un ordre quelconque. On commence par ouvrir une boîte ; ensuite pour chacun des objets on tente de l'ajouter dans l'une des boîtes ouvertes sans dépasser la capacité maximale de celle-ci ; si aucune des boites ne peut le contenir on ouvre une nouvelle boîte.

**Question 1.** *Le nombre de boîtes ouvertes donné par l'algorithme glouton pour des objets dont les poids sont* $8, 5, 3, 7, 3$ *et lorsque la capcité de chaque boîte est de 10.*

On trouve 3 boîtes $\{8, 5\}, \{3, 7\}, \{3\}$ et c'est le nombre minimum de boîtes.

**Question 2.** *Un ordre où l'algorithme glouton donne 4 boîtes alors que l'on peut faire avec 3.*

On trouve 4 boîtes, si on avait considéré les objets dans l'ordre $3, 3, 5, 7, 8$ (c'est pour cela qu'il vaut toujours mieux, dans la vie, effectuer les tâches les plus lourdes d'abord !)

**Question 3.** Ecriture en détail de l'algorithme glouton.

```
nbBoites = 0;
for (i = 0; i < n; ++i){
     k= 0;
     while (k < nbBoites && (contenu[k] + p[i]) > pmax)
            k++;
      if (k == nbBoites) nbBoites++;
      contenu[k] = contenu[k] + p[i];
     }
```

**Question 4.** *Un exemple où l'algorithme glouton donne 4 boîtes alors que l'on peut faire avec 3, et ceci dans le cas où l'on avait considéré les objets dans l'ordre des poids décroissants.*

$$6, 6, 3, 3, 2, 2, 2, 2, 2, 2, 2 \qquad pmax = 10$$

**Question 5.** *Preuve que dans le cas où tous les objets ont un poids strictement supérieur à $pmax/3$ cette deuxième version de l'algorithme glouton donne l'optimum :*

On partage les objets en trois parties, ceux dont le poids est supérieur ou égal à $2pmax/3$, ceux dont le poids est supérieur à $pmax/2$ et inférieur à $2pmax/3$, enfin les autres.

Soit $a, b, c$ le nombre d'objets respectifs dans chacune de ces trois parties. Il est clair que dans toute répartition chaque objet de la première classe sera seul dans sa boîte, deux objets de la deuxième classe ne pourront pas être ensembles dans une boîte et chaque objet de la troisième classe peut être mis éventuellement avec un objet de la deuxième classe et dans tous les cas avec un objet de sa classe. Le nombre $c'$ d'objets de la troisème classe qui seront mis dans une boîte avec un objet de la deuxième est indépendant de l'ordre dans lequel ils sont considérés, par contre il convient de rechercher la boîte ayant la plus petite place disponible pour les mettre, mais ceci est fait par l'algorithme glouton. Le nombre total de boîtes utilisées sera de

$$a + b + \lceil \frac{c - c'}{2} \rceil$$

**Question 6.** *Un exemple où tous les objets ont un poids strictement supérieur à $pmax/3$ et tel que la première version de l'algorithme glouton ne donne pas l'optimum.*

Si les boîtes ont capacité 5 et on considère 4 objets de poids $2, 2, 3, 3$ dans cet ordre, l'algorithme glouton donne trois boîtes : $\{2, 2\}, \{3\}, \{3\}$, alors que si on les avait classés en décroissant on aurait eu 2 boîtes : $\{3, 2\}, \{3, 2\}$