

Programmation Concurrente en $JAVA^{TM}$

Document élaboré par J. Charles et I. Sau.

TP3 : Les tours de Hanoï*

7 Mars 2007

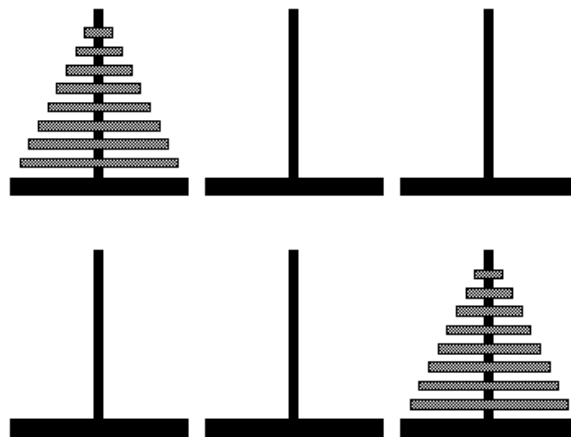
1 Description du problème

Le problème des **tours de Hanoï** est un jeu de réflexion imaginé par le mathématicien français Édouard Lucas (1842-1891) en 1883, et consistant à déplacer des disques de diamètres différents d'une tour de *départ* (tour D) à une tour d'*arrivée* (tour A) en passant par une tour *intermédiaire* (tour I) et ceci en un minimum de coups, tout en respectant les règles suivantes:

- on ne peut déplacer plus d'un disque à la fois,
- on ne peut placer un disque que sur un autre disque plus grand que lui ou sur un emplacement vide.

On suppose que la dernière règle est également respectée dans la configuration de départ.

Voici un exemple avec 8 disques:



Dans ce TP on va programmer un algorithme pour résoudre le problème des tours de Hanoï pour un nombre quelconque de disques.

*Corrigé du TP2: http://www-sop.inria.fr/mascotte/personnel/Ignasi.Sauvalls/Ignasi_Sau.html#teaching

2 Résolution

2.1 Relation de récurrence

Une possible façon de résoudre le problème est en utilisant la notion de **récurrence**:

- Supposons qu'on ait N disques $1, \dots, N$, numérotés de telle façon que si $i > j$ le disque i est plus grand que le disque j .
- A un moment donné, on devra déplacer le disque le plus grand (c'est à dire, le disque numéro N) de la tour D à la tour A . Pour pouvoir faire ce mouvement, les autres $N - 1$ disques devront être bien placés dans la tour I .
- Maintenant le problème consiste à déplacer $N - 1$ disques de la tour I vers la tour A en utilisant la tour D comme tour intermédiaire.

Supposons qu'on dispose d'une méthode *Solve* avec 4 arguments (N et les 3 tours). Alors on peut exprimer cette méthode de la façon suivante:

```
Solve(N, Src, Aux, Dst)
  if N is 0 exit
  Solve(N-1, Src, Dst, Aux)
  Move from Src to Dst
  Solve(N-1, Aux, Src, Dst)
```

En fait, avec cette récurrence on vient de définir explicitement la méthode *Solve*. Par exemple, si on a 3 disques la solution est:

1. Move from Src to Dst
2. Move from Src to Aux
3. Move from Dst to Aux
4. Move from Src to Dst
5. Move from Aux to Src
6. Move from Aux to Dst
7. Move from Src to Dst

Essayez de réfléchir à cet algorithme pendant quelques minutes...

On peut aussi prouver que le nombre de mouvements nécessaires est $2^N - 1$, étant N le nombre de disques.

2.2 Implémentation en Java

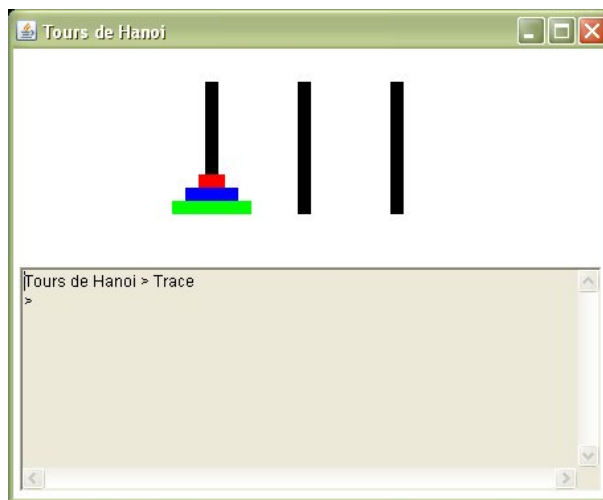
Pour implémenter la solution du problèmes des tours de Hanoi en Java, on va utiliser deux classes principales:

- Une classe *Tour* avec au moins cettes 4 méthodes:
 - void moveTo(Tour dest, Tour tmp, int n)
 - void push(Disque d)
 - Disque pop()
 - Disque head()
- Une classe *Disque* avec au moins cettes 2 méthodes:
 - void goTo(Tour from, Tour to)
 - int size()

3 L'interface graphique

On se propose d'ajouter un interface graphique à l'algorithme des tours de Hanoi. L'interface sera composée de deux éléments:

- Un autre panneau montrant la trace qui avait été générée par l'algorithme au départ
- Un panneau graphique montrant à proprement parler les disques et les pôles et un volet de trace



3.1 Fichier HanoiGUI.java

La classe principale qui devra être ajoutée est la classe HanoiGUI; elle contiendra la méthode main qui configurera et affichera la Frame principale de l'interface graphique. La trace sera inscrite dans un objet de type TextArea. Les disques seront dessinés dans un Canvas.

Pour plus d'information sur l'API, ne pas hésiter à aller voir le site de Sun <http://java.sun.com/j2se/1.5.0/docs/api/>. Les classes nécessaires sont java.awt.Frame, java.awt.Panel, java.awt.TextArea et java.awt.Canvas.

3.2 Les dessins des disques

Fichier PlateauGUI.java Cette classe étend la classe Canvas et sert à dessiner l'ensemble des tours, en surchargeant la méthode `public void paint(Graphics)`. La méthode `public void repaint()` de la classe Canvas servira à mettre à jour l'affichage en cas de modification par l'algorithme.

Fichiers TourGUI.java et DisqueGUI.java Ces deux classes étendent respectivement les classes Tour.java et Disque.java et ont une méthode `public void draw(Graphics g, int x, int y)` qui sert à dessiner les éléments à un emplacement donné.