

# Programmation Avancée et Application

Projet : Partage de butin

---

Jean-Guy Mailly

`jean-guy.mailly@u-paris.fr`

LIPADE - Université de Paris

<http://www.math-info.univ-paris5.fr/~jmailly/>

## Contexte

Après chaque pillage de navire, le capitaine d'un bateau pirate se sert en premier, puis répartit le reste du butin entre les membres de son équipage. Parmi toutes les richesses qu'il n'a pas gardées pour lui, il laisse un objet précieux à chaque pirate. Pour apaiser les tensions sur son navire, il demande à chaque pirate de lui donner ses préférences sur les objets à partager, et il essaye de les respecter au mieux. En plus de leurs préférences, il tient compte des relations au sein de l'équipage. Certains pirates s'apprécient peu, et s'il y a des sources de jalousies entre eux, cela pourrait mener à des disputes ou une mutinerie...

# Premier exemple



- Facile : chaque pirate peut avoir son trésor préféré

## Deuxième exemple



- Problème : on ne peut pas donner 2 à la fois à *A* et *B*
- Si on donne 2 à *A*, *B* est jaloux (et vice-versa), donc on donne 2 à *C*
- Si on donne 3 à *B*, *C* est jaloux, donc on donne 1 à *B* et 3 à *A*

Erratum : *B* est jaloux de *C* dans ce cas, donc pas de solution.

# Troisième exemple



- Problème : on ne peut pas donner 2 à la fois à A et B
- Si on donne 2 à A, B est jaloux (et vice-versa), donc on donne 2 à C
- Si on donne 3 à B, C est jaloux, donc on donne 1 à B et 3 à A
- A et B sont jaloux l'un de l'autre : il n'y a pas de solution !

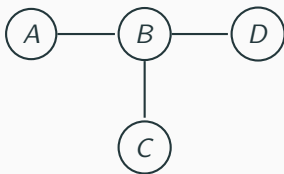
## Sujet

Le capitaine pirate nous demande de l'aider, en développant un logiciel qui permet :

1. de représenter les pirates et les relations entre eux ;
2. de simuler le partage des ressources entre eux ;
3. de calculer le coût d'une solution (le nombre de pirates jaloux), et de le minimiser.

# Modélisation

- On partage  $n$  objets entre  $n$  pirates
- Équipage = graphe non-orienté
- Préférences d'un pirate = liste  $I$  d'objets, telle que  $I[i]$  est strictement préféré à  $I[j]$  si  $i < j$
- Solution : affectation d'un objet à chaque pirate
- Coût = nombre de pirates qui préféreraient avoir l'objet affecté à un de leurs voisins



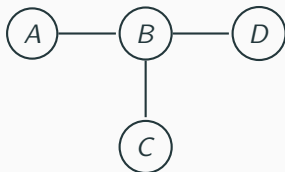
$A : o_1, o_2, o_3, o_4$

$B : o_1, o_3, o_2, o_4$

$C : o_3, o_2, o_1, o_4$

$D : o_1, o_4, o_2, o_3$

## Une solution naïve



$A: o_1, o_2, o_3, o_4$

$B: o_1, o_3, o_2, o_4$

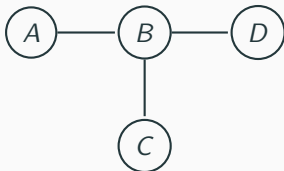
$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible



## Une solution naïve



$A: o_1, o_2, o_3, o_4$

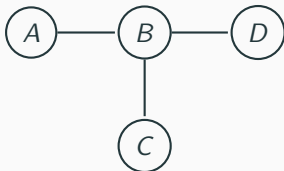
$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible
- $A$  reçoit  $o_1$

# Une solution naïve



$A: o_1, o_2, o_3, o_4$

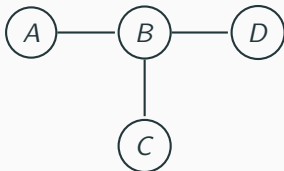
$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible
- $A$  reçoit  $o_1$
- $B$  reçoit  $o_3$

# Une solution naïve



$A: o_1, o_2, o_3, o_4$

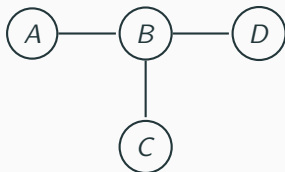
$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible
- $A$  reçoit  $o_1$
- $B$  reçoit  $o_3$
- $C$  reçoit  $o_2$

# Une solution naïve



$A: o_1, o_2, o_3, o_4$

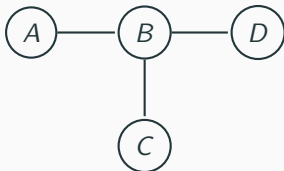
$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible
- $A$  reçoit  $o_1$
- $B$  reçoit  $o_3$
- $C$  reçoit  $o_2$
- $D$  reçoit  $o_4$

# Une solution naïve



$A: o_1, o_2, o_3, o_4$

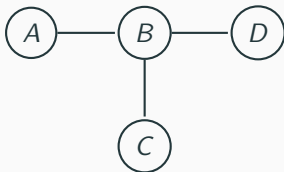
$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- On donne à chaque pirate son objet préféré encore disponible
- $A$  reçoit  $o_1$
- $B$  reçoit  $o_3$
- $C$  reçoit  $o_2$
- $D$  reçoit  $o_4$
- Il y a des jalousies :  $B$  préfère l'objet de son voisin  $A$ ,  $C$  préfère l'objet de son voisin  $B$
- Coût de la solution : 2

## Une solution moins naïve ?



$A: o_1, o_2, o_3, o_4$

$B: o_1, o_3, o_2, o_4$

$C: o_3, o_2, o_1, o_4$

$D: o_1, o_4, o_2, o_3$

- Question : comment trouver une solution optimale ?
  - solution optimale = solution avec le coût le plus petit possible

- Le projet est à réaliser par **groupes de 2 ou 3 étudiants**, issus du même groupe de TD
- Le projet sera à réaliser en plusieurs parties, qui seront déposées sur Moodle
- Les instructions détaillées seront disponibles sur Moodle prochainement