

Séance 8: RÉSEAU SOCIAL

Université Paris Cité

Objectifs:

- Manipulation des tableaux de tableaux d'entiers
- Boucles imbriquées

Nous voulons dans ce TP modéliser un réseau social dans lequel plusieurs utilisateurs ayant les numéros $0, 1, 2 \dots n$ sont inscrits. Pour ce faire, nous allons représenter le réseau par un tableau de tableaux d'entiers R où chaque case $R[i][j]$ vaut :

- 1 si l'utilisateur i est ami avec l'utilisateur j
- 0 sinon.

Par exemple, un réseau R dans lequel il y a trois utilisateurs peut valoir le tableau de tableaux suivant :

```
1 { {0, 1, 0}, {1, 0, 0}, {0, 0, 0} }
```

Où l'utilisateur 0 est ami avec l'utilisateur 1 (car $R[0][1]=1$ et $R[1][0]=1$) et où l'utilisateur 2 n'est ami avec personne.

Notez qu'un utilisateur n'est jamais ami avec lui-même et que pour chaque couple d'utilisateurs i et j : si i est l'ami de j alors j est aussi l'ami de i .

Exercice 1 (Les fonctionnalités de base, * – **)

1. Écrivez une fonction `int[][] CreateGraph (int n)` qui retourne un réseau sous la forme de tableaux de tableaux comme vu ci-dessus. Il y aura dans ce tableau un nombre n d'utilisateurs et l'amitié entre deux utilisateurs est aléatoire. Vous pouvez pour cela utiliser la fonction `randRange` qui vous est donnée au TP5.
2. Écrivez une fonction `int friends_nbr (int[][] R, int a)` qui pour un réseau R et un utilisateur a retourne le nombre d'amis que possède a dans le réseau.
3. Écrivez une fonction `int[] friends (int[][] R, int a)` qui pour un réseau R et un utilisateur a retourne un tableau contenant tous les amis de a dans le réseau et un tableau vide s'il n'en possède aucun.
4. Écrivez une fonction `int[] popular (int[][] R)` qui retourne le tableau contenant tous les utilisateurs les plus populaires du réseau. Un utilisateur est considéré comme populaire s'il possède le plus grand nombre d'amis dans le réseau.
Indication : on pourra d'abord écrire une fonction `int max_friends_nbr (int[][] R)` qui calcule le nombre maximal d'amis parmi les utilisateurs du réseau. Ainsi, un utilisateur est populaire si son nombre d'amis est exactement égal à `max_friends_nbr(R)`.
5. Écrivez une fonction `int[] common_friends (int[][] R, int a, int b)` qui retourne pour un réseau R et deux utilisateurs a et b un tableau contenant les amis qu'ils ont en commun.

- Écrivez une fonction `int[] [] add_user(int[] [] R, int[] t)` qui ajoute un utilisateur dans le réseau `R` et le lie d'amitié avec les utilisateurs qui sont dans le tableau `t`. Cette fonction retourne le nouveau réseau obtenu. Dans le nouveau réseau on aura donc un utilisateur en plus. On supposera que le tableau `t` ne contient que des utilisateurs déjà existants.
- Considérons un tableau `noms` qui contient des chaînes de caractères. Pour chaque indice `i` la case `noms[i]` contient le nom de l'utilisateur `i`. Par exemple, si `noms` vaut :

```
1 noms = {"Evan Spiegel", "Mark Zuckerberg", "Jack Dorsey"}
```

Alors l'utilisateur numéro 1 est `noms[1]` et donc 'Mark Zuckerberg'. Quelle est la longueur de ce tableau ? Modifiez la fonction `popular` pour qu'elle retourne les noms des utilisateurs les plus populaires au lieu de leurs numéros. Attention il faudra bien sûr changer aussi les arguments de la fonction.

□

Exercice 2 (Évènement, ★★★)

Dans cette partie on souhaite utiliser notre réseau social afin d'améliorer l'alchimie de votre pendaison de crémaillère à venir.

- Pour cela vous devez programmer une fonctionnalité `int[] invite(int[] [] R, int a)` qui pour un utilisateur `a` retourne tous ses amis sur le réseau mais aussi les amis de ses amis. Bien entendu cette fonction ne doit pas retourner `a`.
- Certains sont stricts et veulent que chaque invité connaisse deux amis distincts de `a`. Pour vous aider dans cette tâche on vous demandera de calculer le tableau de tableaux `P` où chaque case de `P` est obtenue de `R` de la manière suivante :

$$P[i][j] = \sum_{k=0}^{n-1} (R[i][k] * R[k][j])$$

Où n est le nombre d'utilisateurs du réseau. Une fois `P` calculée, pour chaque couple d'utilisateurs i, j si `P[i][j]=2` alors i connaît deux amis distincts de j .

Écrivez la fonction `int[] strict_invite(int[] [] R, int a)` qui retourne le tableau d'invités selon les conditions strictes.

- Modifiez les fonctions ainsi que la liste de leurs paramètres pour que les valeurs retournées soient les noms des utilisateurs au lieu de leurs numéros.

□