

Récurtivité¹ : Algorithmes et Complexité

Partie 3

Salem BENFERHAT

Centre de Recherche en Informatique de Lens (CRIL-CNRS)
email : benferhat@cril.fr

¹Version préliminaire du cours. Tout retour sur la forme comme sur le fond est le bienvenu.

Récurtivité Croisée

Récursivité croisée

Deux fonctions sont dites mutuellement récursives si elles dépendent les unes des autres.

Baguenaudière



Récurtivité : Baguenaudière

Baguenaudière : principe

On se donne une table (que l'on appellera baguenaudière) à n cases, chacune peut contenir un pion. Chaque case est numérotée de "1" à "n". La baguenaudière peut être vide ou pleine.

Exemple : Une baguenaudière pleine à 5 cases



Baguenaudière : Règles du jeu

- Pour la case 1, on peut enlever un pion ou mettre un pion sans contrainte.
- Pour la case 2, on peut enlever un pion ou mettre un pion uniquement si la case 1 est pleine
- De manière générale, pour la case i (différente de 1 et 2), on peut enlever un pion ou mettre un pion si :
 - La case $i - 1$ est pleine (contient un pion)
 - Toutes les cases de 1 à $i - 2$ sont vides.

Baguenaudière : But du jeu

- La baguenaudière peut être vide ou pleine.
- Si elle est vide, le but est de la remplir
- Si elle est pleine, le but est de la vider

Exemple de baguenaudière

Remplir 5 cases

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1

2

3

4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1

2

3

4

5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3

4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3

4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4

5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : remplir 5 cases

1



2



3



4



5



Exemple de baguenaudière

Vider 5 cases

Déroulement du Jeu Baguenaudiere : vider 5 cases

1

2

3

4

5

Déroulement du Jeu Baguenaudiere : vider 5 cases

1

2

3

4

5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3

4

5

Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4

5

Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5

Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Déroulement du Jeu Baguenaudiere : vider 5 cases

1



2



3



4



5



Exercices

- On rappelle qu'une baguenaudière peut être vide ou pleine.
- Ecrire une fonction récursive qui consiste à remplir une baguenaudière vide.
- Ecrire une fonction récursive qui consiste à vider une baguenaudière pleine.

Récurtivité : Fonction remplir

```
void Remplir (int N)
{
    if (N>0)
    {
        if (N==1) {
            printf("Mettre un pion dans la case %d \n", N);
        }
        else
        {
            Remplir(N-1);
            Vider (N-2);
            printf("Mettre un pion dans la case %d \n", N);
            Remplir(N-2);
        }
    }
}
```

Récurtivité : Fonction vider

```
void Vider (int N)
{
    if (N>0)
    {
        if (N==1) {
            printf( "Enlever un pion de la case %d \n", N);
        }
        else
        {
            Vider (N-2);
            printf("Enlever un pion de la case %d \n", N);
            Remplir(N-2);
            Vider(N-1);
        }
    }
}
```

Nouvelle variante

On considère de nouveau le problème de la tour de Hanoï, mais en ajoutant une contrainte supplémentaire :

- On ne peut échanger de disques entre la tige de départ et la tige intermédiaire

Tour de Hanoï

Une nouvelle variante

Contrainte supplémentaire

Aucun disque ne peut être échangé entre la tige de départ et la tige intermédiaire

Tour de Hanoï

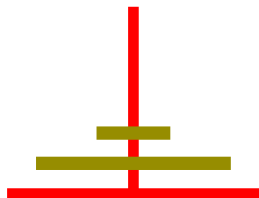
Une nouvelle variante

Variante de Hanoi

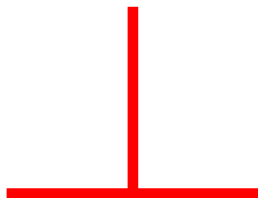
Nombre de disques = 2

Tour de Hanoï

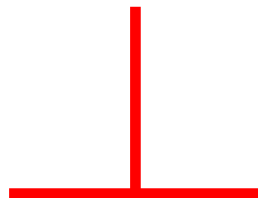
Etape 0: La tour à l'état initial



Tige 0



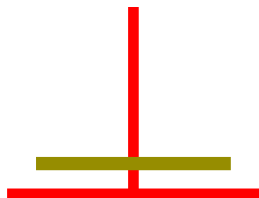
Tige 1



Tige 2

Tour de Hanoï

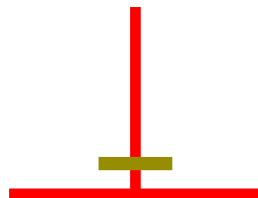
Etape 1: déplacer un disque de la tige 0 vers la tige 2



Tige 0



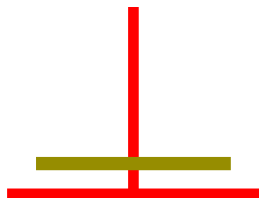
Tige 1



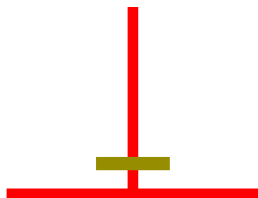
Tige 2

Tour de Hanoï

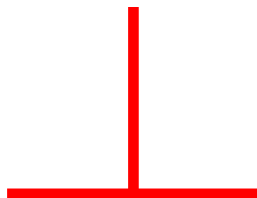
Etape 2: déplacer un disque de la tige 2 vers la tige 1



Tige 0



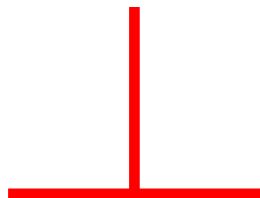
Tige 1



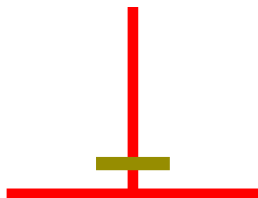
Tige 2

Tour de Hanoï

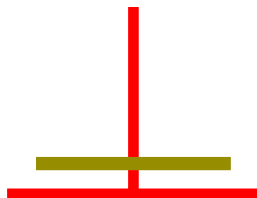
Etape 3: déplacer un disque de la tige 0 vers la tige 2



Tige 0



Tige 1



Tige 2

Tour de Hanoï

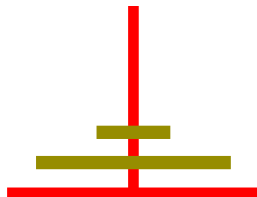
Etape 4: déplacer un disque de la tige 1 vers la tige 2



Tige 0



Tige 1



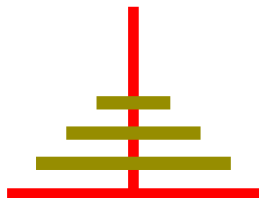
Tige 2

Variante de Hanoi

Nombre de disques = 3

Tour de Hanoï

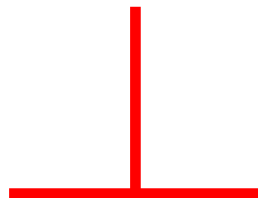
Etape 0: La tour à l'état initial



Tige 0



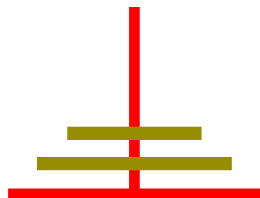
Tige 1



Tige 2

Tour de Hanoï

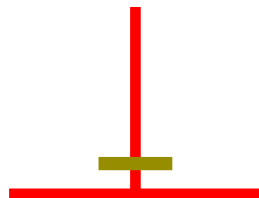
Etape 1: déplacer un disque de la tige 0 vers la tige 2



Tige 0



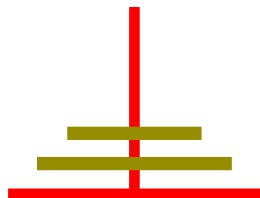
Tige 1



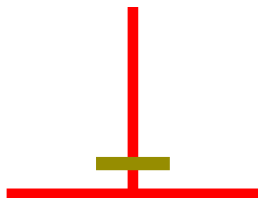
Tige 2

Tour de Hanoï

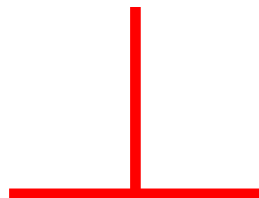
Etape 2: déplacer un disque de la tige 2 vers la tige 1



Tige 0



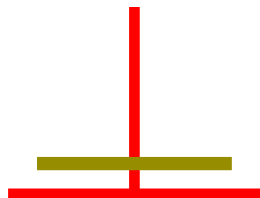
Tige 1



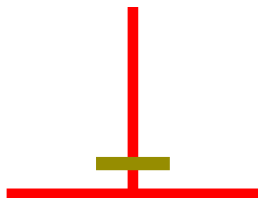
Tige 2

Tour de Hanoï

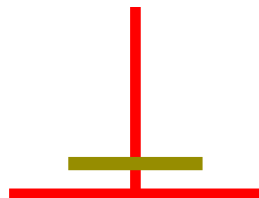
Etape 3: déplacer un disque de la tige 0 vers la tige 2



Tige 0



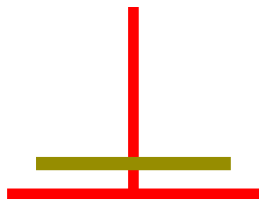
Tige 1



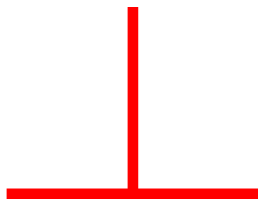
Tige 2

Tour de Hanoï

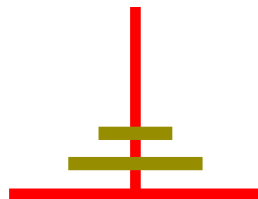
Etape 4: déplacer un disque de la tige 1 vers la tige 2



Tige 0



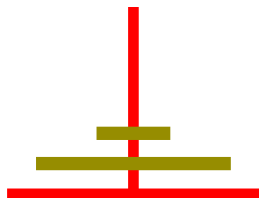
Tige 1



Tige 2

Tour de Hanoï

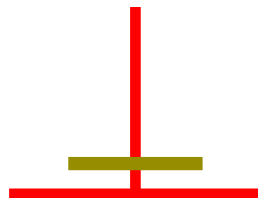
Etape 5: déplacer un disque de la tige 2 vers la tige 0



Tige 0



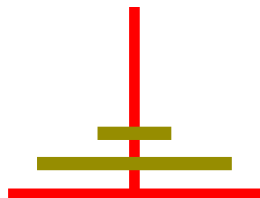
Tige 1



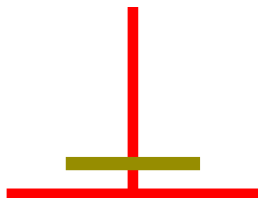
Tige 2

Tour de Hanoï

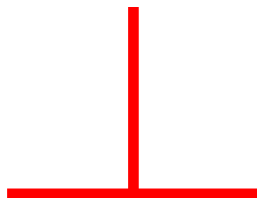
Etape 6: déplacer un disque de la tige 2 vers la tige 1



Tige 0



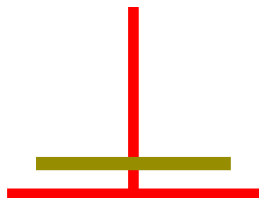
Tige 1



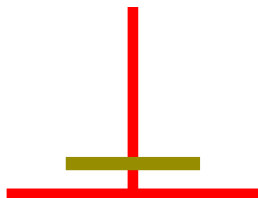
Tige 2

Tour de Hanoï

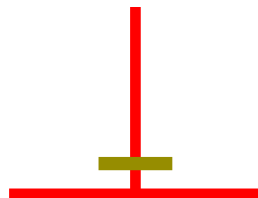
Etape 7: déplacer un disque de la tige 0 vers la tige 2



Tige 0



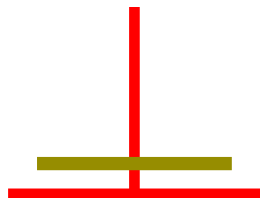
Tige 1



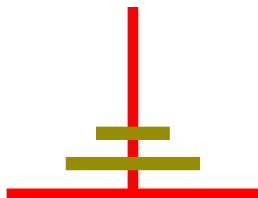
Tige 2

Tour de Hanoï

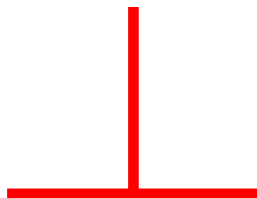
Etape 8: déplacer un disque de la tige 2 vers la tige 1



Tige 0



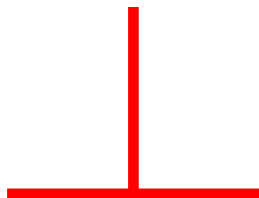
Tige 1



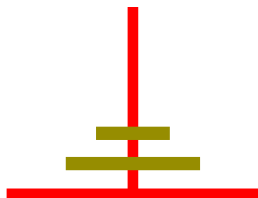
Tige 2

Tour de Hanoï

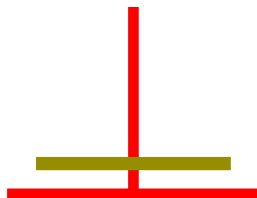
Etape 9: déplacer un disque de la tige 0 vers la tige 2



Tige 0



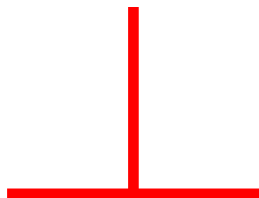
Tige 1



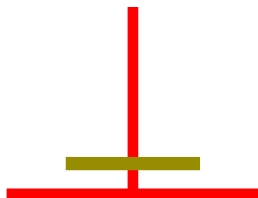
Tige 2

Tour de Hanoï

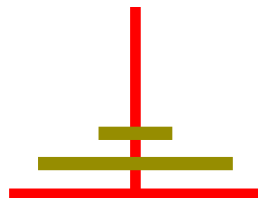
Etape 10: déplacer un disque de la tige 1 vers la tige 2



Tige 0



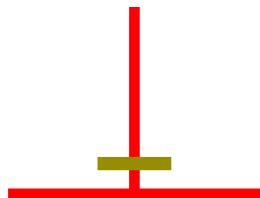
Tige 1



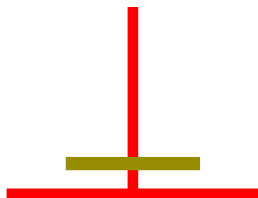
Tige 2

Tour de Hanoï

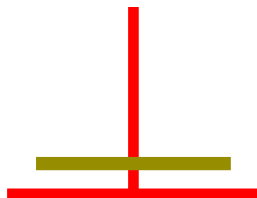
Etape 11: déplacer un disque de la tige 2 vers la tige 0



Tige 0



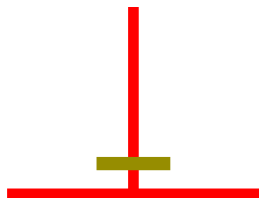
Tige 1



Tige 2

Tour de Hanoï

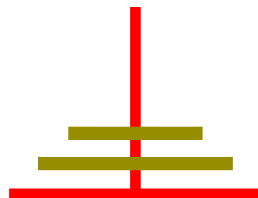
Etape 12: déplacer un disque de la tige 1 vers la tige 2



Tige 0



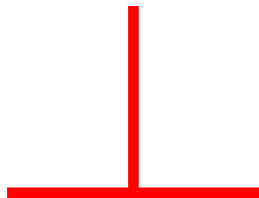
Tige 1



Tige 2

Tour de Hanoï

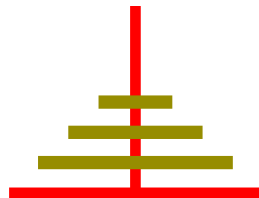
Etape 13: déplacer un disque de la tige 0 vers la tige 2



Tige 0



Tige 1



Tige 2

Récurtivité croisée : variante de la tour de Hanoï

Cette variante de la tour de Hanoï a une solution naturelle avec la récursivité croisée, avec les deux fonctions suivantes :

Première fonction

Hanoi_sans_AB (n,A,B,C)

qui consiste à transférer n disques de la tige A vers la tige C sans utiliser de communications entre A et B (dans les deux sens).

Première fonction

Attention à l'utilisation des arguments:

- l'appel à la fonction `Hanoi_sans_AB (n-1,A,B,C)` permet de transférer $(n-1)$ disques de la tige A vers la tige C sans utiliser de communications entre A et B (tout va bien).
- l'appel à la fonction `Hanoi_sans_AB (n-1,B,A,C)` permet de transférer $(n-1)$ disques de la tige B vers la tige C sans utiliser de communications entre A et B (tout va bien).
- **mais** l'appel à la fonction `Hanoi_sans_AB (n-1,C,A,B)` permet de transférer $(n-1)$ disques de la tige C vers la tige A sans utiliser de communications entre A et C (et non pas entre A-B). Donc il est important de tenir compte des paramètres lorsque vous écrivez votre fonction.

Deuxième fonction

Hanoi_sans_BC (n,A,B,C)

qui consiste à transférer n disques de la tige A vers la tige C sans utiliser de communications entre B et C (dans les deux sens).

Exercice

Supposons que les deux fonctions suivantes sont disponibles :

`Hanoi_sans_AB (n-1,A,B,C)`

`Hanoi_sans_BC (n-1,A,B,C)`

- Ecrire les fonctions (réversives) `Hanoi_sans_AB (n,A,B,C)` et `Hanoi_sans_BC (n,A,B,C)`
- Evaluer leur complexité

Variante de Hanoi

Solutions

Réversibilité croisée : Hanoi sans AB

```
void hanoi_sans_dep_inter
    (int nbre_disques, int dep, int inter, int dest)
{
    if(nbre_de_disques!=0)
    {
        hanoi_sans_dep_inter(nbre_disques-1,dep,inter,dest);
        hanoi_sans_inter_arrive(nbre_disques-1,dest,dep,inter);
        printf("Deplacer un disque de %d vers %d\n",dep,dest);
        hanoi_sans_dep_inter(nbre_disques-1,inter,dep,dest);
    }
}
```

Réversivité croisée : Hanoi sans BC

```
void hanoi_sans_inter_arrive
    (int nbre_disques, int dep, int inter, int dest)
{
    if(nbre_de_disques!=0)
    {
        hanoi_sans_inter_arrive(nbre_disques-1,dep,dest,inter);
        printf("Deplacer un disque de %d vers %d\n",dep,dest);
        hanoi_sans_dep_inter(nbre_disques-1,inter,dest,dep);
        hanoi_sans_inter_arrive(nbre_disques-1,dep,inter,dest);
    }
}
```