



Systèmes Concurrents

TP PROBLEME DES PHILOSOPHES

Deuxième Année, Systèmes Logiciels

YOUNES SAOUDI

2020-2021

Les Fourchettes Sont des Ressources Critiques

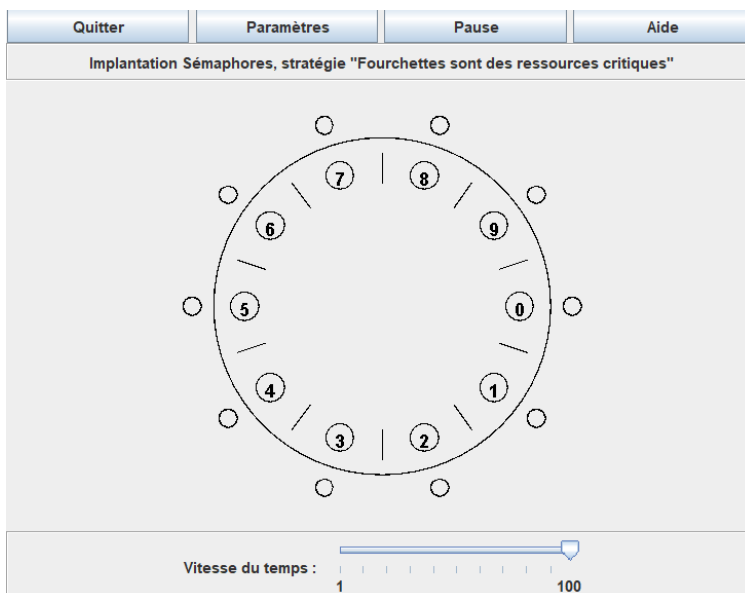
1. Version de base :

```
public void demanderFourchettes (int no) throws InterruptedException
{
    availableForks[Main.FourchetteDroite(no)].acquire();
    //Thread.sleep(2000);
    availableForks[Main.FourchetteGauche(no)].acquire();
}

/** Le philosophe no rend les fourchettes.
 * Précondition : il possède les deux fourchettes adjacentes à son assiette.
 * Postcondition : il n'en possède aucune. Les fourchettes peuvent être libres ou réattribuées à un autre philosophe.
 */
public void libererFourchettes (int no)
{
    availableForks[Main.FourchetteDroite(no)].release();
    availableForks[Main.FourchetteGauche(no)].release();
}
```

Voir Implémentation PhiloSemBase.java

2. Mise en Evidence de l'Interblocage : Il suffit de décommenter la ligne Thread.sleep(2000) entre les prises de fourchettes :



3. Solutions évitant l'interblocage :

- Rupture de la symétrie :** Il suffit que le dernier philosophe se comporte différemment des autres en prenant la fourchette de gauche en premier puis celle de droite au lieu de l'inverse :

```

public void demanderFourchettes(int no) throws InterruptedException {
    if (no == Main.nbPhilosophes - 1) {
        availableForks[Main.FourchetteGauche(no)].acquire();
        availableForks[Main.FourchetteDroite(no)].acquire();
    } else {
        availableForks[Main.FourchetteDroite(no)].acquire();
        availableForks[Main.FourchetteGauche(no)].acquire();
    }
}

```

Voir Implémentation PhiloSemBaseSolution.java

- b. **Limitation des ressources** : On peut aussi utiliser un sémaphore avec `nbPhilosophes - 1` jetons :

```

public void demanderFourchettes(int no) throws InterruptedException {
    philosophersEating.acquire();
    availableForks[Main.FourchetteDroite(no)].acquire();
    availableForks[Main.FourchetteGauche(no)].acquire();
}

/**
 * Le philosophe no rend les fourchettes. Précondition : il possède les deux
 * fourchettes adjacentes à son assiette. Postcondition : il n'en possède
 * aucune. Les fourchettes peuvent être libres ou réattribuées à un autre
 * philosophe.
 */
public void libererFourchettes(int no) {
    availableForks[Main.FourchetteDroite(no)].release();
    availableForks[Main.FourchetteGauche(no)].release();
    philosophersEating.release();
}

```

Voir Implémentation PhiloSemBaseSolution2.java

Progression d'un Philosophe en Fonction de l'état de Ses Voisins

Voir Implémentation PhiloSemEtatsVoisins.java

```

Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 2
Nombre de philosophes qui mangent en même temps : 2
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 1
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 2
Nombre de philosophes qui mangent en même temps : 1
Nombre de philosophes qui mangent en même temps : 2
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 2
Nombre de philosophes qui mangent en même temps : 3
Nombre de philosophes qui mangent en même temps : 3

```

Figure 1 - Degré de parallélisme de la première approche (10 philosophes)

Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 5
 Nombre de philosophes qui mangent en même temps : 3
 Nombre de philosophes qui mangent en même temps : 3
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4
 Nombre de philosophes qui mangent en même temps : 4

Figure 2 - Degré de parallélisme de la seconde approche (10 philosophes)

Nous pouvons déduire des figures 1 et 2 que la deuxième approche offre plus de parallélisme que la première.

Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 22
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 23
 Nombre de philosophes qui mangent en même temps : 24

Figure 3 - Degré de parallélisme de la première approche (100 philosophes)

Nombre de philosophes qui mangent en même temps : 36
 Nombre de philosophes qui mangent en même temps : 37
 Nombre de philosophes qui mangent en même temps : 38
 Nombre de philosophes qui mangent en même temps : 38
 Nombre de philosophes qui mangent en même temps : 39
 Nombre de philosophes qui mangent en même temps : 40
 Nombre de philosophes qui mangent en même temps : 42
 Nombre de philosophes qui mangent en même temps : 38
 Nombre de philosophes qui mangent en même temps : 40
 Nombre de philosophes qui mangent en même temps : 40
 Nombre de philosophes qui mangent en même temps : 39
 Nombre de philosophes qui mangent en même temps : 38
 Nombre de philosophes qui mangent en même temps : 38
 Nombre de philosophes qui mangent en même temps : 36
 Nombre de philosophes qui mangent en même temps : 36

Figure 4 - Degré de parallélisme de la seconde approche (100 philosophes)

Nous pouvons déduire des figures 3 et 4 que le degré de parallélisme de la première approche est plus susceptible à diminuer en fonction du nombre de philosophes que la seconde approche.

Equité

Dans le cas de 4 philosophes, si deux opposés effectuent toutes leurs actions plus rapidement que les deux autres, ces derniers ne pourront jamais manger et seront en famine.

Cette famine est évitée en utilisant *Fair Semaphores* ; c.à.d l'**option FIFO** des sémaphores, imposant ainsi **un ordre de priorité** aux processus.

Etude :

1. Degré de parallélisme au pire des cas :

Dans le pire de cas, les philosophes mangent un par un à tour de rôle suivant leur ordre dans la file des sémaphores, contrairement au cas optimal où presque 40% des philosophes mangent en même temps.

2. Attente maximale pour un philosophe demandeur :

L'attente maximale est de $\text{nbPhilosophes} - 1$.

3. Limites de la solution :

Cette solution est limitée car le degré de parallélisme est inférieur à celui de la solution optimale.

Voir Implémentation `PhiloSemEquite.java`