

Problème des cinq philosophes

Document d'analyse de réalisation du projet

Rendu du projet

Voici les différents éléments constituant le rendu de ce projet :

- les sources du programme, stockées dans le dossier **sources/** ;
- le programme compilé en « .jar », disponible dans le dossier courant, à lancer avec la commande suivante : « java -jar philosophes.jar » ;
- le document présent permettant de constituer un petit « lisez-moi » et d'expliquer le programme.

Explication du problème

Ce problème, plus communément nommé « le dîner des philosophes », pose un problème majeur de partage de ressource en informatique, et c'est ce qui rend le projet intéressant dans le contexte actuel de l'enseignement réalisé.

En effet, on se trouve confronté au problème suivant : cinq philosophes se trouvent autour d'une table et chacun a devant lui un plat de nourriture. Étant en pleine pensée philosophique, les philosophes ont besoin d'énergie, et ils ont faim. Le problème étant que pour manger, ils ont besoin de deux baguettes, une à gauche et une à droite de chacun d'entre eux : elles sont donc partagées (chacune pour deux philosophes). La gestion des actions des philosophes doit donc être réfléchie et gérée correctement pour représenter au mieux le problème et comprendre comment le résoudre.

Réalisation du programme

Le programme que j'ai réalisé n'est qu'une **simple représentation** du problème posé. D'autant plus qu'il en existe de nombreuses variantes. Afin de rendre cette représentation facile à comprendre, j'ai tenu à m'imposer la réalisation d'un espace graphique (IHM) simple et épuré pour comprendre le problème rapidement : j'ai donc réalisé une table ainsi qu'un espace en forme de cercle. Les philosophes sont représentés par des disques de couleurs différentes suivant leur état, et les baguettes par des segments blancs. Il est possible de choisir un nombre de philosophes directement depuis l'interface afin de réaliser des tests et de comprendre le problème plus facilement.

Techniquement parlant, j'ai utilisé des sémaphores en JAVA pour faire fonctionner mon programme : chaque philosophe est en réalité un sémaphore, dont voici les principales étapes :

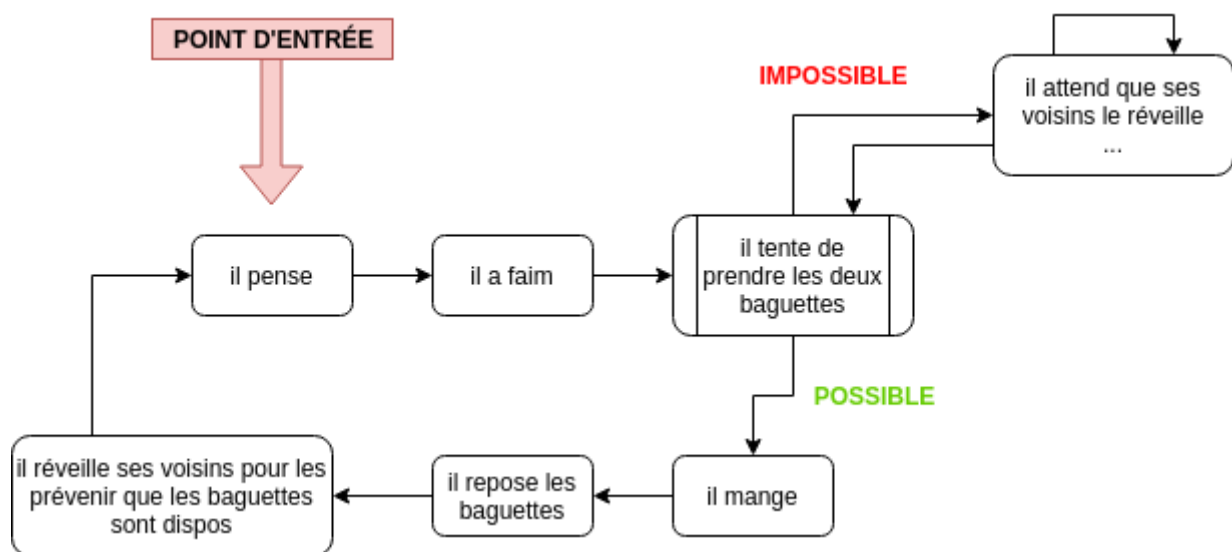


Diagramme simplifié du fonctionnement de mon programme

J'ai préféré ce comportement à celui de créer un sémaphore par baguette pour éviter de gérer un cas assez embarrassant : si une baguette veut être prise par deux philosophes au même moment, il faudra bien faire un choix. Faut-il préférer un philosophe plutôt que l'autre ? Comment met-on cela en place ? Il aurait par exemple fallu une action supplémentaire pour chaque philosophe : reposer une des deux baguettes. Mon choix s'est donc porté sur la facilité de conception pour la mise en place des sémaphores.