



Projet UNIX étrangement avancé

Philosophes

42 staff staff@42.fr

Résumé: Ce projet est pour vous la découverte des threads et des mutex.

Table des matières

I	Préambule	2
II	Les Philosophes mangeurs de riz	3
III	Infos pratiques	5

Chapitre I

Préambule

Les papous :

Chez les papous, y'a des papous papa et des papous pas papa...
Mais certains sont à poux et d'autres sans poux....

Donc chez les papous :

y'a des papous papa

y'a des papous pas papa

y'a des papous papa à poux

y'a des papous papa pas à poux

y'a des papous pas papa à poux

y'a des papous pas papa pas à poux

Chez les poux,

y'a des poux papa

y'a des poux pas papa

Donc chez les papous,

y'a des papous papa à poux papa

y'a des papous papa à poux pas papa

y'a des papous papa pas à poux papa

y'a des papous papa pas à poux pas papa

y'a des papous pas papa à poux papa

y'a des papous pas papa à poux pas papa

y'a des papous pas papa pas à poux papa

y'a des papous pas papa pas à poux pas papa

Si vous avez tout compris, bravo, c'est que vous savez ce qu'est du binaire, un parcours d'arbre, de la récursivité,... .

Chapitre II

Les Philosophes mangeurs de riz

C'est l'histoire très connue de sept philosophes.

Comme tous les philosophes, ils se posent des questions. Cette fois-ci, ils se sont mis en tête de ne prendre leurs repas qu'avec une et une seule baguette par personne. Ils sont convaincus qu'avec leur esprits habiles ils sauront se partager suffisamment rapidement les baguettes pour ne pas laisser l'un d'entre eux mourir de faim. Le cerveau d'un philosophe consomme en effet beaucoup de sucre. Ils doivent donc manger très régulièrement pour ne pas tomber en hypoglycémie.

Si vous n'avez pas réussi à les convaincre de renoncer à leur petit défi, en revanche, vous avez pu les persuader d'utiliser un simulateur de votre conception plutôt que de risquer leur santé.

Maintenant, leur survie ne dépend que de vous.

Le simulateur devra répondre au cahier des charges suivants :

- 7 philosophes devront être assis autour d'une table.
- Chaque philosophe possède un thread qui lui est propre.
- À la gauche de chaque philosophe se trouve une baguette (il y a donc 7 baguettes en tout).
- Un philosophe ne peut se saisir d'une baguette que si elle se trouve adjacente à lui (droite ou gauche).
- Deux philosophes ne peuvent pas utiliser la même baguette au même moment (paraît-il pour des raisons d'hygiène).
- Un philosophe passe par 3 étapes :
 - Mange avec les 2 baguettes (droite et gauche).
 - Repos (ne mange pas) et ne détient aucune baguette (même si celle-ci est

disponible).

- Réfléchit, dans ce cas le philosophe a une et une seule baguette en main.
- Après l'état repos, un philosophe peut soit manger, soit réfléchir. Néanmoins, si un voisin réfléchit déjà, il aura la priorité pour acquérir la baguette commune.
- Un philosophe qui ne mange pas perd 1 point de vie par SECONDE.
- Après qu'un philosophe ait mangé, il récupère toute sa vie, et passe en Repos.
- Un philosophe dont les points de vie tombent à 0 meurt. À cet instant la simulation s'interrompt jusqu'à l'appui d'une touche (dont le choix vous appartient). Cet appui provoque la libération des ressources et l'arrêt complet du programme.
- Afin de faciliter la soutenance, vous devrez avoir défini dans votre projet les macros suivantes :
 - MAX_LIFE : Le nombre de points de vie maximum des philosophes.
 - EAT_T : Le nombre de SECONDES que met un philosophe à manger.
 - REST_T : Le nombre de SECONDES pendant lesquels un philosophe se repose.
 - THINK_T : Le nombre de SECONDES pendant lesquels un philosophe réfléchit.
 - TIMEOUT : Le temps en SECONDES après lesquels la simulation s'interrompt en affichant "Now, it is time... To DAAAAAAAAANCE!!!" si aucun philosophe n'est mort de faim (le dîner en cours doit rester visible à l'écran pour faire le bilan). L'appui d'une touche provoque la libération des ressources et l'arrêt complet du programme.
- Vous devez rendre une interface graphique (mlx, SDL, ncurses, OpenGL, ...) illustrant clairement ce qui se passe dans le programme.

Chapitre III

Infos pratiques

- Ce projet est à réaliser en binôme c'est à dire 2 et pas plus.
- Ce projet ne sera corrigé que par des humains. Vous êtes donc libres d'organiser et nommer vos fichiers comme vous le désirez, en respectant néanmoins les contraintes listées ici.
- L'exécutable devra se nommer **philo**
- Vous devez coder en C sur un dump standard et rendre un Makefile avec les règles habituelles.
- Si vous êtes malin et que vous utilisez votre bibliothèque **libft**, vous devez en copier les sources et le **Makefile** associé dans un dossier nommé **libft** qui devra être à la racine de votre dépôt de rendu. Votre **Makefile** devra compiler la librairie, en appelant son **Makefile**, puis compiler votre projet.
- Selon la librairie graphique que vous utilisez, elle peut être incluse dans votre rendu, et comme la libft elle doit être compilée automatiquement par le Makefile principal.
- Votre projet doit être à la Norme. C'est la norminette qui fait foi.
- Vous devez gérer les erreurs de façon raisonnée. En aucun cas votre programme ne doit quitter de façon inattendue (Segmentation fault, etc...).
- Vous devez rendre, à la racine de votre dépôt de rendu, un fichier **auteur** contenant vos logins suivis d'un '\n' chacun :

```
$> cat -e auteur
xlogin$
ylogin$
$>
```

- Dans le cadre de votre partie obligatoire, vous avez le droit d'utiliser les fonctions suivantes :
 - malloc.
 - free.
 - write.

- `pthread_create`.
 - `pthread_detach`.
 - `pthread_join`.
 - `pthread_mutex_init`.
 - `pthread_mutex_destroy`.
 - `pthread_mutex_trylock`.
 - `pthread_mutex_lock`.
 - `pthread_mutex_unlock`.
 - Vous avez l'autorisation d'utiliser d'autres fonctions dans le cadre de vos bonus, à condition que leur utilisation soit justifiée lors de votre soutenance. Il y a bien sûr des points pour les bonus.
- Vous pouvez poser vos questions sur le forum ...
 - Bon courage à tous !