

Mécanisme asynchrone

Imaginons une tâche très lente. Une interface utilisateur digne de ce nom se doit de matérialiser la progression de cette tâche sous une forme ou une autre (barre de progression graphique, affichage de points, etc.). Cet exercice a pour but de simuler une telle situation.

Le programme que l'on demande de rédiger a la syntaxe suivante :

`async` *durée-en-ms*

La tâche lente est implémentée sous forme d'un thread qui progresse de 1 % toutes les d millisecondes ; tant que l'avancement n'est pas de 100 %, ce thread attend la durée d spécifiée avec `usleep`, puis il incrémente l'avancement. Plus la durée d est faible, plus la tâche progresse vite.

Un autre thread simule l'interface graphique : tant que l'avancement n'atteint pas 100 %, il affiche l'avancement actuel (celui que la tâche met à jour), puis attend une seconde. On notera donc que l'avancement est une variable partagée par les deux threads.

Le thread principal crée les 2 threads, puis attend qu'ils se terminent.

Par exemple :

```
> ./async 27
0 %
36 %
73 %
termine
```

intervalle de 1 seconde entre chaque affichage

le thread principal indique que tout est terminé

Vous n'utiliserez bien sûr aucune variable globale.