

Projets CPE : Python Concurrent

Réalisation d'un système multi-tâches de simulation d'un restaurant

Difficulté : **/*****

On considère le système (*temps réel*) simple suivant qui :

1. simule des commandes de clients dans un restaurant
2. un certains nombre de serveurs en salle enregistrent ces commandes et les transmettent à la cuisine pour préparation
3. après leur préparation, les serveurs délivrent ces commandes aux clients

Dans la **version de base**, on n'identifie pas de cuisinier et ce sont les serveurs qui simulent la préparation des commandes (voir plus bas pour la version étendue).

Prévoir :

- s processus *serveur*. P. Ex. $s = 5$
- un processus *clients* qui simulera aléatoirement les commandes des clients selon une loi uniforme. Ce processus émettra une commande aléatoire toutes les p. ex. 3..10 *secondes* à l'adresse des serveurs.
- un processus *major_dHomme* qui s'occupera des affichages à l'écran
- un tampon de taille (p. ex.) 50 contiendra les commandes des clients ; les serveurs prélèvent des commandes de ce tableau
- une commande d'un client sera constituée d'un identifiant client (un entier) et une lettre $A..Z$ qui représentera le menu commandé

En l'absence d'interface graphique, on utilisera le module **curses** de Python que l'on a déjà utilisé dans l'exemple cours de chevaux. On affichera ainsi à l'écran les informations suivants :

- les commandes des clients (les paires $(id, menu)$) dès leur émission
- le serveur qui prend cette commande en charge et simule sa préparation (par un délai)
- le client qui reçoit sa commande préparée

☞ Les informations sont affichées exclusivement par le processus *major_dHomme*.

Un exemple d'affichage à l'écran :

Le serveur 1 traite la commande (id_i, C_i) (ou rien si pas de commande traité par ce serveur)

....

Le serveur s traite la commande (id_j, C_j)

Les commandes clients en attente : $[(id_i, C_i), (id_j, C_j) \dots (id_k, C_k)]$

Nombres de commandes attente : 5

Commande (id_u, U) **est servie au client**

Aller plus loin (Bonus) :

Ajouter un certains nombre de cuisiniers (en cuisine) qui préparent ces commandes et avertissent les serveurs. Le serveur qui avait enregistré la commande la délivre au client qui a commandée.

Ajouter à l'aversion de base :

- c processus *cuisto*. P. Ex. $c = 2$
- Modifier les affichage et présenter le cuisinier qui traite la commande.

Le contenu de l'écran sera augmenté des lignes :

Le cuisinier 1 prépare la commande $(id_1, A, serveur_1)$ (ou rien si pas de commande traité par ce cuisinier)

....

Le cuisinier c prépare la commande $(id_p, P, serveur_p)$
