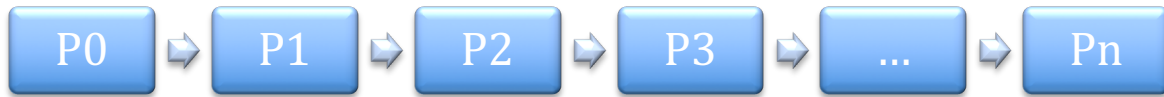


Mini projet de Système d'Exploitation

L3 MIAGE filière apprentissage

On souhaite créer une architecture de processus en « pipeline » tel que le processus P_i ait un fils P_{i+1} avec un tube en commun : le père écrit sur le tube et le fils lit sur ce tube



On souhaite utiliser cette architecture pour filtrer les nombres non premiers. Pour cela chaque processus P_i « garde » un nombre premier Z_i (2, 3, 5, 7, 11, etc.).

Le comportement des processus est le suivant :

- Le processus P_0 lit des nombres au clavier et s'ils ne sont pas pairs les envoie au processus P_1 (son fils) ;
- Le processus P_1 reçoit des nombres de son père et s'ils ne sont pas multiples de 3 il les envoie à son fils (P_2)
- De façon générale le processus P_{i+1} reçoit des nombres de la part de son père P_i sur le tube qu'ils partagent et si le nombre reçu n'est pas multiple de Z_i (le nombre premier « gardé » par P_i) il le transmet à son fils sinon il ne fait rien et attend un autre nombre de son père.
- Le dernier processus P_N affiche les nombres au lieu de les passer à son fils (il est le dernier) lorsque le nombre reçu de son père n'est pas multiple de Z_N .

On suppose de plus que l'on possède un fichier Prime.txt qui contient un entier N puis les $N+1$ premiers nombres premiers (ex si $N=4$, le fichier contiendra 4 puis 2 3 5 7 11).

Écrire la procédure (fonction void) qui lit les données de ce fichier et construit un tableau T_Prime qui stockera les $N+1$ nombres premiers du fichier Prime.txt et définit une variable globale N égale à la taille de ce tableau (dans l'exemple précédent le tableau T_Prime sera égal à [2, 3, 5, 7, 11] et N vaudra 4).

En réutilisant ce que vous avez écrit précédemment écrire le programme complet qui

- Lit le fichier T_Prime et initialise N ;
- Crée (et démarre) les $N+1$ processus P_0 à P_N ;
- Lit des entiers au clavier (processus P_0) ;
- Termine lorsque l'on saisit le nombre 0 (avec la terminaison de tous les fils)

Vous ferez une première version avec 1 seul processus ; une seconde avec 2 puis enfin vous réaliserez la version finale à N processus (N étant une constante du programme, ou mieux passée en paramètre du programme).