

# Test de Connaissances de Systèmes d'Exploitation 2011

## Introduction

On doit écrire un programme C qui utilise trois autres processus fils H, M et S qui incrémentent les trois aiguilles d'une horloge. Comme leur nom l'indique, chacun des trois processus s'occupe respectivement de l'heure, la minute et la seconde. C et H sont reliés avec deux pipes, et les processus communiquent également avec des signaux.

## Structure

Le processus C attend un signal `SIGQUIT` (du clavier, par exemple) et alors demande au processus H l'heure, lui envoyant d'abord le signal `SIGUSR1` et puis un entier sur un pipe pour expliquer à H ce qu'il attend (un entier en retour, ou bien une chaîne de caractères). Ensuite bien entendu C lit la réponse de H depuis l'autre pipe le reliant à H.

Les processus S, M et H communiquent entre eux avec des signaux ainsi.

Le processus S dérouté `SIGALARM` et met en place une alarme à chaque seconde écoulée, tout en comptant les secondes. Lorsque le compteur de secondes dépasse 59, il le remet à zéro, et envoie un signal `SIGUSR1` au processus M.

Le processus M incrémente le compteur de minutes à chaque réception de signal `SIGUSR1`. Lorsque le compteur de minutes dépasse 59, il le remet à zéro, et envoie un signal `SIGUSR2` au processus H.

Le processus H incrémente le compteur d'heures à chaque réception de signal `SIGUSR2`. Lorsque le compteur d'heures dépasse 23, il le remet simplement à zéro.

Comme dit plus haut, le processus H, lors de la réception d'un signal `SIGUSR1`, lit d'abord depuis le pipe le reliant à C l'entier codifiant le type de renseignement : 1 pour les entiers, 2 pour les chaînes de caractères, et ensuite écrit sur l'autre pipe le reliant à C la valeur du compteur d'heures, dans le format souhaité. Pour le format chaînes de caractères, notre protocole fera que le processus H écrit d'abord le nombre de caractères qu'il s'apprête à envoyer, et ensuite il écrit ces caractères.

## Travail à faire

1. Ces processus ont-ils des sections critiques ? Ont-ils besoin d'être protégés contre les signaux ?
2. Si on fait que les traitants de signaux incrémentent directement les compteurs, quel sera le type spécial de données utilisé pour déclarer les compteurs ?
3. On comprend que C devra se dupliquer plusieurs fois pour créer les autres processus. L'ordre dans lequel il le fait a-t-il une importance ? (Indication :

réfléchissez au renseignement essentiel dont a besoin tout processus pour envoyer un signal à un autre processus).

4. L'horloge ainsi conçue avec ces processus pourrait-elle être exacte (c'est-à-dire aussi exacte que la vraie horloge matérielle du système) ? Donnez-en deux raisons expliquées clairement.
5. Faites un schéma (graphique) représentant les processus et leurs communications, avec des flèches, etc.
6. Écrivez d'abord le code source en lignes générales, en laissant de l'espace pour les détails de chaque processus. Autrement dit : écrivez d'abord les instructions communes, à effectuer avant toute duplication (descripteurs de pipe, etc.). Écrivez ensuite les duplications, et dans chaque processus mettez simplement des lettres A, B, C, D pour indiquer leur corps à écrire plus tard. Écrivez enfin les traitants de signal et les déclarations de compteurs, etc. (qui seront en fait à mettre tout au début du programme). N'oubliez pas de prendre soin de zombies et faites attention à ce qu'on ne fasse pas de « petits lapins » non plus.
7. Uniquement une fois que l'architecture vous semble bien claire, commencez à détailler le corps du programme.
8. Comment pourrait-on étendre ce programme afin que C affiche non seulement la valeur du compteur d'heure, mais aussi les minutes et les secondes ? Décrivez-en brièvement la structure (au plus cinq lignes de texte).