

## TP6

### 1 Job control

Sous Linux, il existe l'application **xeyes** qui affiche des yeux.

**Question 1** : Lancez cette application et interrompez-la à l'aide d'un signal **SIGINT** envoyé depuis le clavier (en utilisant `^c`).

**Question 2** : Lancez à nouveau ce script en arrière plan. À l'aide des commandes vues en cours, faites le passer au premier plan, arrêtez-le. Vérifiez sa présence à l'aide de la commande **ps**. Finalement, redémarrez-le et tuez-le à l'aide de la commande **kill** et du signal **SIGKILL**

Lancer alors maintenant plusieurs fois cette application en arrière plan (vous pouvez faire en sorte qu'elles aient des couleurs différentes à l'aide de l'option `-fg`).

**Question 3** : En utilisant la commande **kill** et les numéros de job, envoyer alors le signal **SIGSTOP** à une des instances. Que se passe-t-il ?

**Question 4** : Analyser le comportement de cette application à l'envoi des signaux : **SIGUSR1**, **SIGQUIT**, **SIGTERM**.

**Question 5** : Fermez alors le shell dans lequel vous aviez lancé les applications. Que se passe-t-il ? Relancez une nouvelle application qui "survive" à la fermeture du shell.

### 2 Trap

Le but des questions suivantes est d'implémenter un timer modifiable à l'aide de signaux et de la commande **sleep**.

**Question 6** : En utilisant la commande **sleep**, écrire un script qui prend un argument et le décrémente toutes les secondes puis affiche un message après

**Question 7** : Modifier le programme pour qu'il affiche le temps restant lorsqu'il reçoit le signal **SIGUSR1**.

**Question 8** : Ajouter également une fonctionnalité qui retarde le compteur de 10 secondes si le signal **SIGUSR2** est reçu.

**Question 9** : Lancez le script puis stoppez-le. Que se passe-t-il ? Et si vous lancez ensuite le signal **SIGUSR1** ?