Projet 2 - NSY103

- Vous allez réutiliser la bibliothèque **biblio.c**

1) Vous allez développer un programme **projet2.c** qui créé un processus fils :

Le processus père se présente : **<numero de processus> : Je suis le pere**

Le processus fils se présente : **<numero de processus> : Je suis le fils**

Le processus fils invite l'utilisateur à saisir un message :

<numero de processus> : Veuillez saisir un message

L'utilisateur saisit un message, le fils transmet le message par UDP (transmission vers adresse IP **127.0.0.1** et numéro de port **8000**) à son père et le père répond :

<numero de processus> : Message reçu : <message>

* Attention pour que le père reçoive le message il faut qu'il soit à l'écoute (en réception sur le port **8000**) avant que le fils n'envoie le message.

2) Vous allez créer un programme **projet2bis.c**, il s'agira du programme **projet2.c** dans lequel vous rajouterez une boucle du côté du père et une boucle du côté du fils afin que la séquence saisie / transmisison / affichage du message reçu se répète sans fin.

Vous allez sans doute constater à l'utilisation que les affichages du père et du fils se télescopent dans votre terminal, parce que lorsque le fils envoie le message au père il affiche immédiatement

<pid fils> : Veuillez saisir un message

Et ensuite le père juste après réception du message affichera

<pid pere> : Message reçu : <message>

Avant que l'utilisateur n'ai eu le temps de saisir son message

Pour éviter ce télescopage le fils juste après avoir envoyé un message va s'interrompre lui-même avec le signal **SIGSTOP** (cf. fonction **kill()** sur internet, analogue à la commande **kill** que nous avons déjà utilisée).

Et le père juste après avoir reçu un message va relancer le fils avec le signal **SIGCONT**.

Si vous avez des problèmes de compréhension de l'énoncé du projet n'hésitez pas à poser vos questions sur le Forum.