TP 2 : Gestion des processus

## Exercice 1 :

### Question 1

**top** est utilisé de manière interactive/dans un terminal et que **ps** est plutôt utilisé dans les scripts, combiné avec d'autres commandes bash

Pstree : Afficher un arbre des processus

Question 2 :

Ps -a

* **R** en cours d'exécution.
* **T** processus stoppé.
* **I** processus endormi (>20s).
* **S** processus endormi (<20s).
* **Z** processus zombie.
* **D** processus non interruptible.
* **W** processus swappé sur disque.

## Exercice 2 :

### Question 1 :

Chacun de ces sous-répertoires contient des fichiers et des sous-répertoires exposant des informations sur le processus avec l'ID du processus correspondant.

### Question 2 :

ps -o ppid=

# Exercice 3 :

max\_threads = mempages / (8 \* THREAD\_SIZE / PAGE\_SIZE);

#define PAGE\_SIZE (AC(1, UL) << PAGE\_SHIFT)

#define THREAD\_SIZE (2\*PAGE\_SIZE)

Exercice 4 :

Ps -le

Question 2 :

cat /proc/PID/stat | awk '{print "priority " $18 " nice " $19}'

Question 3 :

On remarque que la commande ps -l est exécuté avec une priorité de 5.

Question 4 :

sleep 240 &nice sleep 240 &ps -l

On remarque que le nice sleep augmente la priorité du second sleep de 10. En effet le premier a une priorité de 5 et le second une priorité de 15.

Question 5 : renice +5 PID

Exercice 5 :

text data bss dec hex filename

2285 632 4194336 4197253 400b85 a.out

Question 2 :

On constate que tout se passe dans le ‘heap’ et pas dans le ‘stack’