

TD1 : Création de processus.

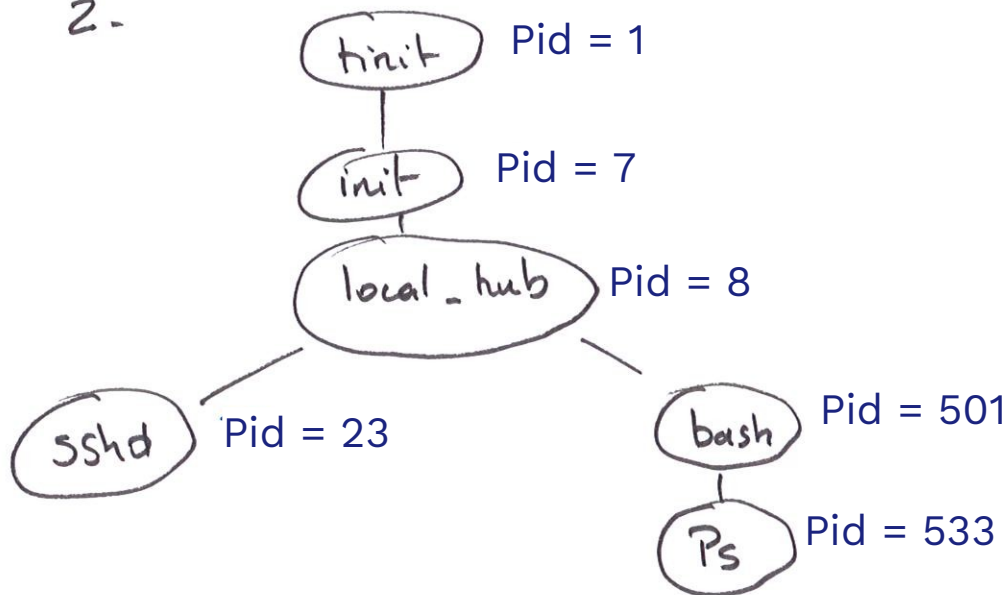
Exercice 1

1. `getty` → `PID=2` : créé après `init` → `PID=1`
`PPID=1` : `PID` du père → `init`

~\$ ps -aef

UID	PID	PPID	C	STIME	TTY	TIME	CMD
user	1	0	0	17:50	?	00:00:00	/cocalc/bin/tini -v -g -- sh -c env -i /cocalc/init/init.sh \$COCA
user	7	1	0	17:50	?	00:00:00	sh -c env -i /cocalc/init/init.sh \$COCALC PROJECT ID \$KUCALC IMAG
user	8	7	22	17:50	?	00:00:05	node --optimize-for-size --gc-interval=100000 --always-compact /c
user	23	8	0	17:50	?	00:00:00	sshd: /usr/sbin/sshd -D -p 2222 -h /tmp/.cocalc/ssh host rsa key
user	501	8	0	17:51	pts/0	00:00:00	/bin/bash
user	533	501	0	17:51	pts/0	00:00:00	ps -aef

2.



Exercice 2

1. Des lignes de 111... 111 alternent avec des lignes de 222... 22

On voit que les processus sont exécutés tour à tour dans un ordre qui n'est pas régulier \rightarrow le nombre de lignes de 111... 11 ainsi que celles de 222 n'est pas régulier

2. Avec la commande `ps -aef` on voit que

`/bin/bash . /test1.sh \rightarrow PID = 1037`
`. /test2.h \rightarrow PID = 1038` $\downarrow + 1$

Le père de ces processus est `/bin/bash`

(PID = 921)

Exercice 3

1. Les valeurs s'incrémentent de 1 à chaque ligne par contre le processus qui écrit la valeur n'est pas toujours le même.

Chaque processus possède ses propres ressources. Ici le fichier est partagé pour chaque processus. Par contre chaque processus possède son propre pointeur de fichier. Il apparaît donc dans le fichier le PID du dernier processus qui a écrit la ligne.

