

Notion de processus : le vocabulaire

- **Code source** : Écriture dans un langage de programmation
- **Interprétation** : Analyse du code source par la machine pour pouvoir exécuter un programme.
- **Compilation** : Transformation du code source en fichier exécutable qui est le programme.
- **Fichier exécutable / Programme** : Ensemble d'instructions destiné à être exécuté par le processeur. Un programme est difficile à comprendre pour les humains, mais ne pose aucun problème pour la machine

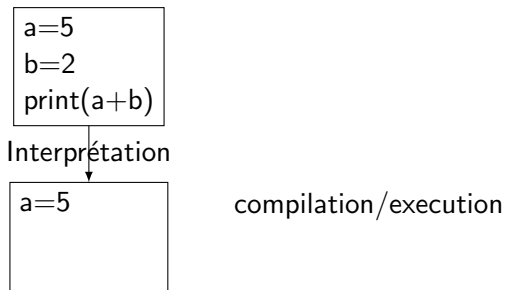
Notion de processus : Compilation vs Interprétation

- **Interprétation** : Analyse du code source par la machine pour pouvoir exécuter un programme.
- **Compilation** : Transformation du code source en fichier exécutable qui est le programme.

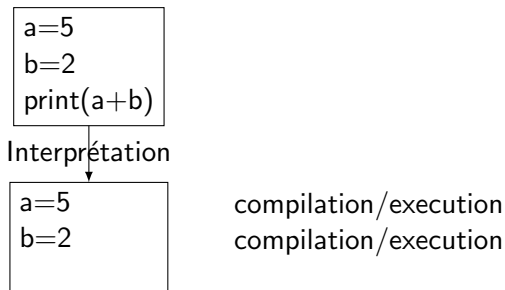
Un langage interpréteur : le python

```
a=5  
b=2  
print(a+b)
```

Un langage interpréteur : le python



Un langage interpréteur : le python



Un langage interpréteur : le python

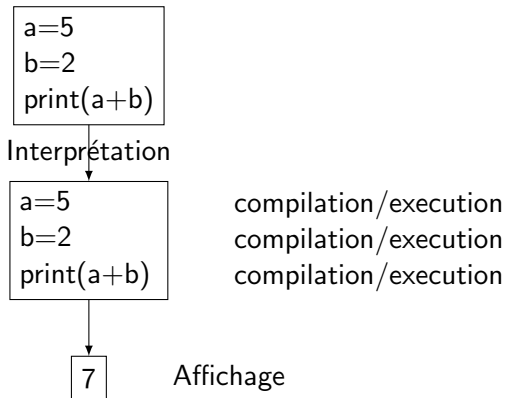
```
a=5  
b=2  
print(a+b)
```

Interprétation

```
a=5  
b=2  
print(a+b)
```

compilation/execution
compilation/execution
compilation/execution

Un langage interpréteur : le python



Un langage compilé : le C

```
int main(){  
    int a,b;  
    a=5;  
    b=2;  
    printf("%f",a+b);} 
```

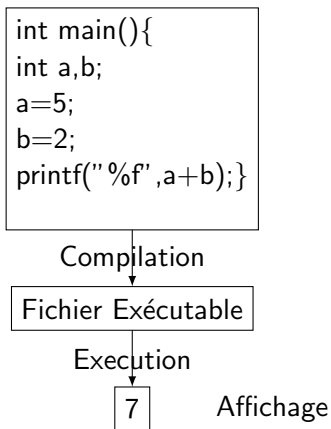

Un langage compilé : le C

```
int main(){  
  int a,b;  
  a=5;  
  b=2;  
  printf("%f",a+b);} 
```

Compilation

Fichier Exécutable

Un langage compilé : le C



Notion de processus

Définition

Un processus est une instance d'un programme, chargée en mémoire et qui est exécutée de manière continue ou discontinue par le système d'exploitation.

Il ne faut pas confondre programme et processus. Les états possibles d'un processus sont :

- prêt
- élu
- bloqué
- terminé (Fin)

Notion de processus

- Sous Linux, on peut voir l'état des processus en cours avec la commande *top*;
On peut simuler un terminal (console) linux sur <https://replit.com>
- Sous Windows, on peut voir les processus en cours en faisant *Ctrl+Alt+Suppr* puis *Gestionnaire des taches*.

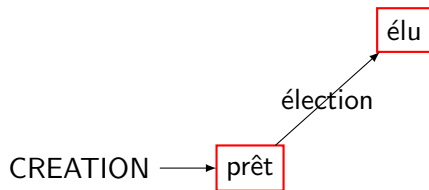
Notion de processus

CREATION

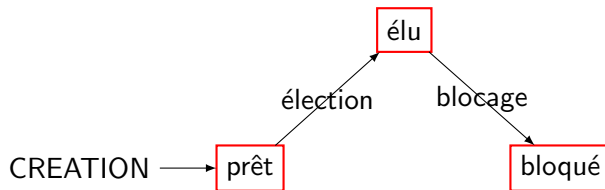
Notion de processus

CREATION → prêt

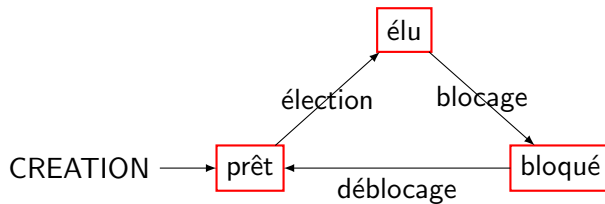
Notion de processus



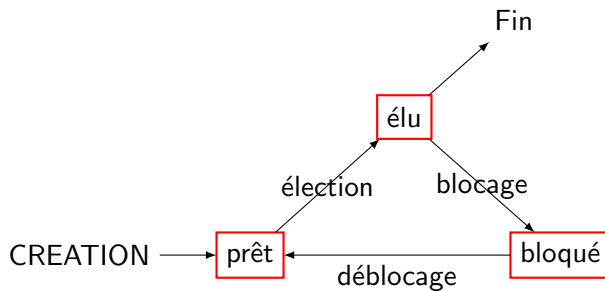
Notion de processus



Notion de processus



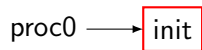
Notion de processus



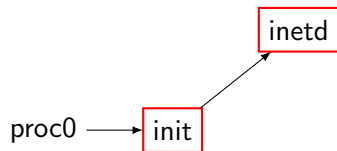
Chaine de Procesus

proc0

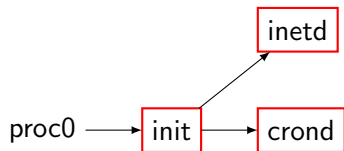
Chaine de Procesus



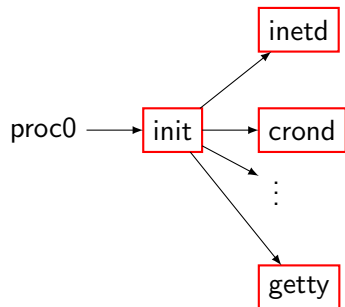
Chaine de Procesus



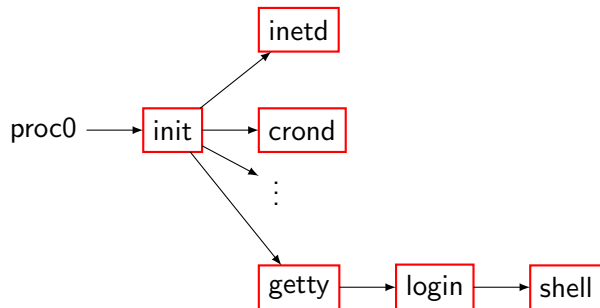
Chaine de Procesus



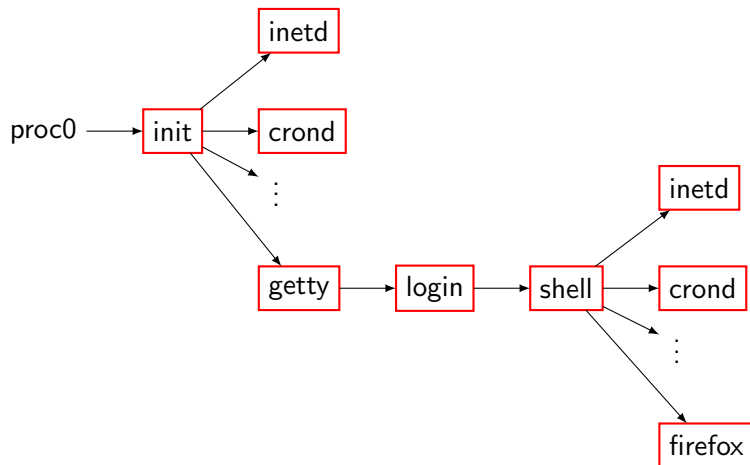
Chaine de Processus



Chaine de Procesus



Chaine de Processus



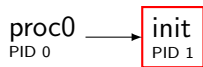
Chaine de Procesus

PID (Process Identification)

proc0
PID 0

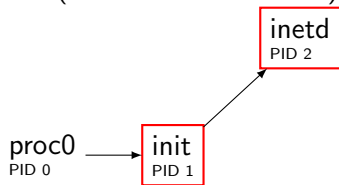
Chaine de Procesus

PID (Process Identification)



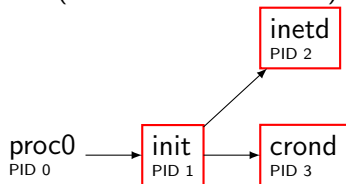
Chaine de Procesus

PID (Process Identification)



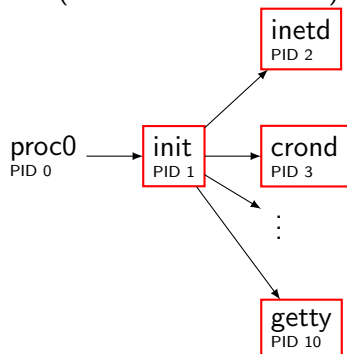
Chaine de Procesus

PID (Process Identification)



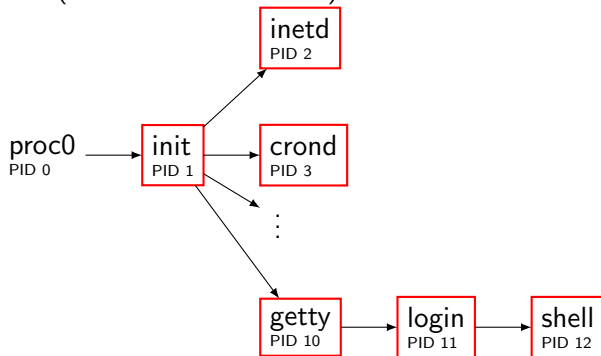
Chaine de Procesus

PID (Process Identification)



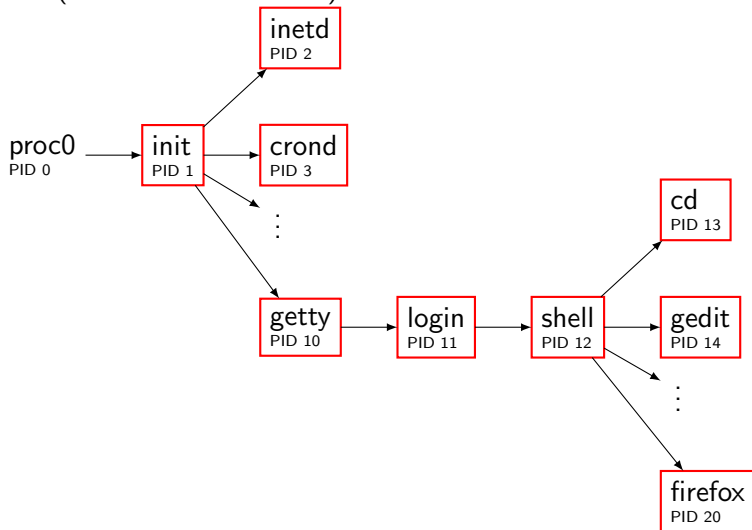
Chaine de Procesus

PID (Process Identification)



Chaine de Processus

PID (Process Identification)



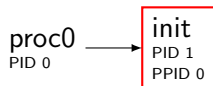
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)

proc0
PID 0

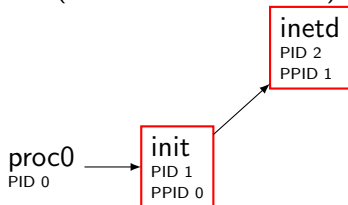
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



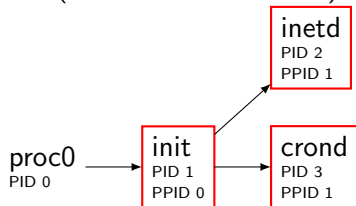
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



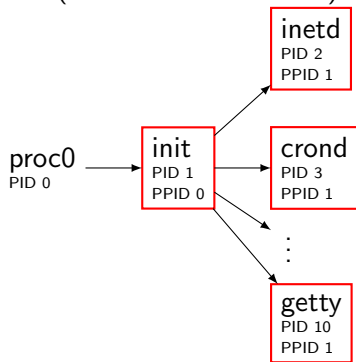
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



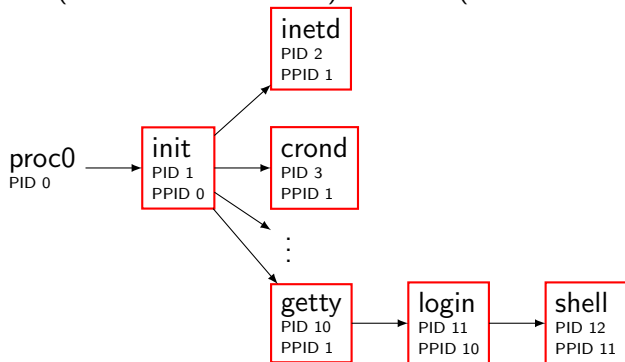
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



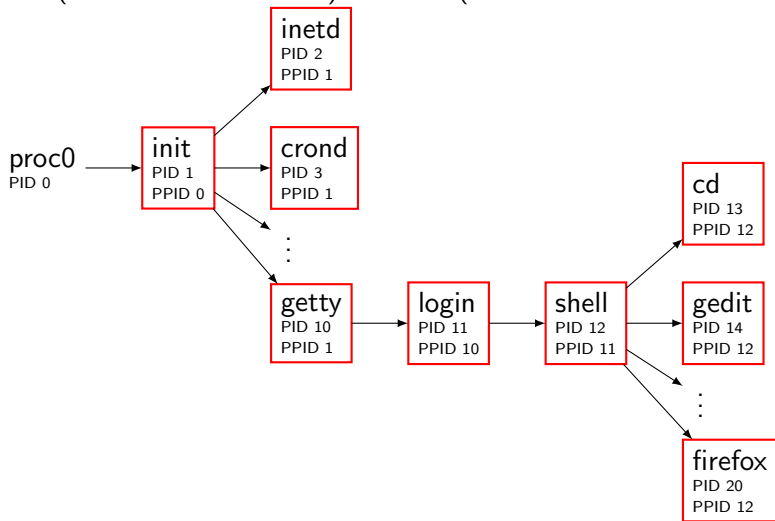
Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



Chaine de Procesus

PID (Process Identification) et PPID (Parent Process Identification)



PID, PPID, SID, PGID, UID

Sous linux, avec la commande :

```
ps xao pid,ppid,pgid,sid,uid,comm|head
```

On peut afficher les informations suivantes :

- PID - ID de Processus
- PPID - Parent ID de Processus
- PGID - ID du Groupe de Processus
- SID - ID de Session
- UID de l'Utilisateur ID
- comm, nom du processus

Interblocage (Deadlock en anglais) : Exemple

On a 1 exercice:

Tracer un cercle et son diamètre.

Un élève 1 possède le compas (R1) et un élève 2 possède la règle (R2).

Interblocage (Deadlock en anglais) : Exemple

On a 1 exercice:

Tracer un cercle et son diamètre.

Un élève 1 possède le compas (R1) et un élève 2 possède la règle (R2).

L'élève 1 fait l'exercice

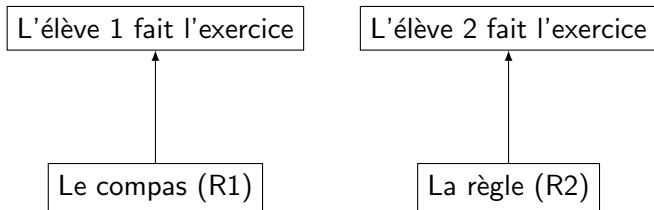
L'élève 2 fait l'exercice

Interblocage (Deadlock en anglais) : Exemple

On a 1 exercice:

Tracer un cercle et son diamètre.

Un élève 1 possède le compas (R1) et un élève 2 possède la règle (R2).

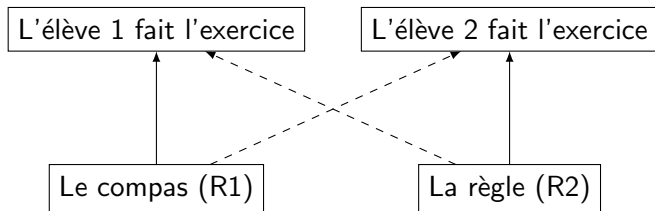


Interblocage (Deadlock en anglais) : Exemple

On a 1 exercice:

Tracer un cercle et son diamètre.

Un élève 1 possède le compas (R1) et un élève 2 possède la règle (R2).

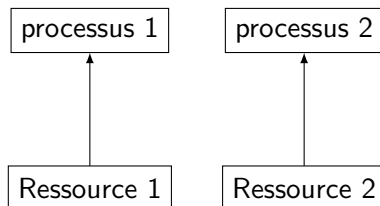


Interblocage (Deadlock en anglais) :Schéma

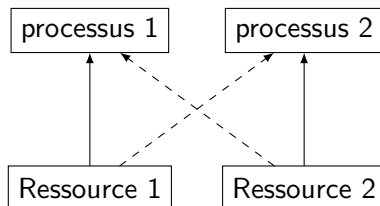
Interblocage (Deadlock en anglais) :Schéma



Interblocage (Deadlock en anglais) :Schéma



Interblocage (Deadlock en anglais) :Schéma



Interblocage (Deadlock en anglais) :Algorithme

Deux algorithmes sont connus pour éviter l'interblocage :

- Utilisation de mutex

https://fr.wikipedia.org/wiki/Exclusion_mutuelle;

- Algorithme du Banquier.

https://fr.wikipedia.org/wiki/Algorithme_du_banquier