

Exercice 1 :

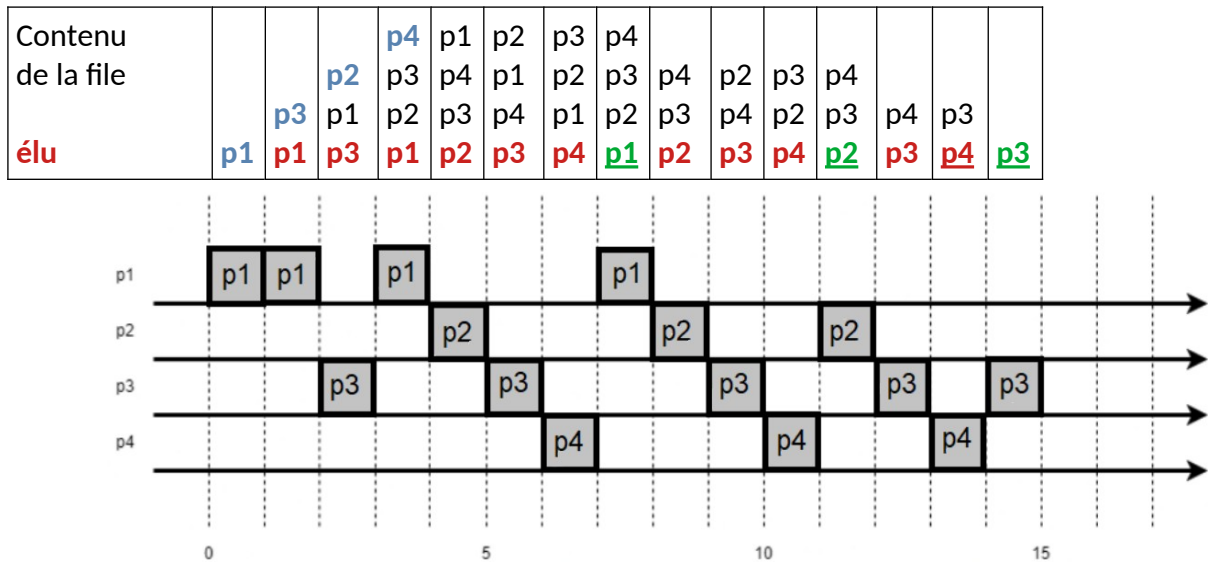
1. Les trois états possibles d'un processus sont : **prêt** (en attente d'un quantum de temps processeur), **bloqué** (en attente de l'accès à une ressources) et **élu** (en cours d'exécution).

2. Dans ce contexte, les deux seuls états possibles sont **prêt** et **élu**.

3.

```
def defile(self):
    if self.est_vide():
        return None
    return self.contenu.pop(0)
```

4. [Attention, un processus élu est prioritaire sur processus créé pour l'ajout dans la file d'après l'exemple de chronogramme donné.]



5. Code complété :

```
class Ordonnanceur:
    def __init__(self):
        self.temps = 0
        self.file = File()

    def ajoute_nouveau_processus(self, proc : Processus):
        ''' Ajoute un nouveau processus dans la file de l'ordonnanceur. '''
        self.file.enqueue(proc)
```

CORRECTION – 2024 Amérique Nord Jour 1

```
def tourniquet(self):
    ''' Effectue une étape d'ordonnancement et renvoie le nom
        du processus élu.'''
    self.temps += 1
    if not self.file.est_vide():
        proc = self.file.defile()
        proc.execute_un_cycle()
        if not proc.est_fini():
            self.file.enfile(proc)
        return proc.nom
    else:
        return None
```

6.

```
def gestion_processus(liste_proc, depart_proc):
    tps = 0
    ordo = Ordonnanceur()
    ordo.ajoute_nouveau_processus(depart_proc[tps])
    while not ordo.file.est_vide():
        proc = ordo.tourniquet()
        print(proc)
        tps += 1
        if tps in depart_proc:
            ordo.ajoute_nouveau_processus(depart_proc[tps])
```

7.

	fichier	clavier	processeur	port internet
D: acquérir le fichier	bloqué par D			
D: faire des calculs	bloqué par D			
B: acquérir le clavier	bloqué par D	bloqué par B		
D: acquérir le clavier (bloqué par B)	bloqué par D	bloqué par B		
A: acquérir le GPU	bloqué par D	bloqué par B	bloqué par A	
B: acquérir le fichier (bloqué par D)	bloqué par D	bloqué par B	bloqué par A	
C: acquérir le port	bloqué par D	bloqué par B	bloqué par A	bloqué par C
A: faire des calculs	bloqué par D	bloqué par B	bloqué par A	bloqué par C
C: faire des calculs	bloqué par D	bloqué par B	bloqué par A	bloqué par C
A: libérer le GPU	bloqué par D	bloqué par B		bloqué par C
C: libérer le port	bloqué par D	bloqué par B		

La situation d'interblocage intervient avec les deux processus D et B qui se bloquent mutuellement en réservant respectivement le fichier et le clavier.

[Par contre les processus A et C peuvent se terminer.]