

Polynésie – 2023 – sujet1 - Correction

Exercice 2 (3 points)

1.

a. Ordre d'exécution avec un quantum de 1 cycle CPU

Données :

PID	Durée
11	4
20	2
32	3

Quantum = 1

Chaque processus exécute 1 cycle puis retourne en fin de file s'il n'est pas terminé.

Déroulement :

11, 20, 32,
11, 20, 32,
11, 32,
11

Suite complète :

11, 20, 32, 11, 20, 32, 11, 32, 11

👉 **Commentaire :** *On tourne tant que des processus ont encore des cycles à exécuter. Le tourniquet répartit équitablement le CPU.*

b. Ordre d'exécution avec un quantum de 2 cycles CPU

Quantum = 2

- 11 exécute 2 cycles (reste 2)
- 20 exécute 2 cycles (terminé)
- 32 exécute 2 cycles (reste 1)
- 11 exécute 2 cycles (terminé)
- 32 exécute 1 cycle (terminé)

Suite des PID :

11, 11, 20, 20, 32, 32, 11, 11, 32

👉 **Commentaire :** *Le quantum plus grand réduit le nombre de rotations dans la file.*

2.

a. Création de la liste d'attente initiale

```
liste_attente = [Processus(11,4), Processus(20,2), Processus(32,3)]
```

👉 **Commentaire :** *L'ordre dans la liste correspond à l'ordre d'arrivée.*

b. Méthodes de la classe Processus

execute_un_cycle

```
def execute_un_cycle(self):
    """Met à jour le reste à faire après l'exécution d'un cycle."""
    self.reste_a_faire = self.reste_a_faire - 1
```

👉 **Commentaire :** *On décrémente le nombre de cycles restants.*

change_etat

```
def change_etat(self, nouvel_etat):
    """Change l'état du processus avec la valeur passée en paramètre."""
    self.etat = nouvel_etat
```

👉 **Commentaire :** *Méthode simple d'affectation d'attribut.*

est_termine

```
def est_termine(self):
    """Renvoie True si le processus est terminé, False sinon."""
    return self.reste_a_faire == 0
```

👉 **Commentaire :** *Un processus est terminé lorsque le nombre de cycles restants est nul.*

c. Complétion de la fonction tourniquet

Code complété :

```
def tourniquet(liste_attente, quantum):
    ordre_execution = []

    while liste_attente != []:

        # On extrait le premier processus
        processus = liste_attente.pop(0)
        processus.change_etat("En cours d'exécution")
        compteur_tourniquet = 0

        while compteur_tourniquet < quantum and not
processus.est_termine():

            ordre_execution.append(processus.pid)
            processus.execute_un_cycle()
            compteur_tourniquet = compteur_tourniquet + 1

        if not processus.est_termine():
            processus.change_etat("Suspendu")
            liste_attente.append(processus)
        else:
            processus.change_etat("Terminé")

    return ordre_execution
```

👉 **Commentaire :**

La boucle interne simule le quantum.

Si le processus n'est pas terminé après le quantum, il retourne en fin de file.

Sinon, il quitte définitivement le système.
