PARALLELISATION MAXIMAL AUTOMATIQUE

Projet de Systèmes d'exploitation

TOUATI Feriel : L3 ASR | SI KADDOUR Mustapha : L3ASR | IBOVI OBOURA Antoine : L3 MIAGE FI

2021/2022

Genèse du Projet

Objectif: Création d'un système permettant d'exécution un Système de tâches valides en utilisant la parallélisation de tâches.

Brainstorming:

- Sémaphore
- Threading
- Multiprocessing
- Graphviz
- Networkx
- Matplotlib

Techniques choisies:

THREADING

Graphviz

Organisation du travail

01.

Brainstorming

La première séance de projet

02.

Documentation

Sur les différents points du projet et les techniques possibles

03.

Choix de la technique

Threading: adéquat et facile à comprendre
Selection des librairies necessaires

04.

Implémentation des classes de base

Classes Task et TaskSystem
Les méthodes simples: getDependencies, constructeurs

05.

Pseudo codes des méthodes difficiles

Telles que Run, Validation des Entrées et draw

06.

Implémentation des méthodes

Implémentation des méthodes et la validation du code avec des tests

Travail d'équipe

Répartition des taches

Réunions: Call Discord Ou en TD

Fusion et correction des Codes

Difficultés rencontrées

- Le choix de la technique
- L'implémentation du parallélisme
- Manipulation des threads
- Dessin du graphe d'exécution en parallèle
- Charge de travail

Structure du code

Classes	Attributs	Méthodes

Task	`name = ""  | reads = []`
	`writes = []| run = None`
	`(fonction)`	

TaskSystem	`listtask = []| precedences ={}`	`getDependencies(task)`
		`Run`
		`Draw()`
		`verification()`
		`Interferences()`

Étapes de parallélisme

ÉTAPE 1

Récupérer les taches sans dépendances

ÉTAPE 2

Lancer les tâches en parallèle en lançant des threads prenant en target les fonctions de ses tâches

ÉTAPE 3

Attendre la fin de l'exécution de tous les threads lancés

ÉTAPE 4

Récupérer les tâches ayant comme dépendances toutes les tâches précédemment exécutées

```
for task in tasktobexcuted:  #this loop is used to launch all tasks ready to be exectuted with threads t

    t = threading.Thread(target=task.run) #define a thread t for each task
    t.start() #run the thread
    threads.append(t) #stock the thread in the threads list
    taskexcuted.append(task) #put the task in the list of tasks who are already launched
    if task in tasknoorder: tasknoorder.remove(task) #remove the task that is excuted from tasknoorder i

#Once the loop is done with executing tasks which were ready to be executed we empty the list tasktobeex
tasktobexcuted = []

 #this loop while is used to wait for the threads to finish their execution (wait till all threads are r
while len(list(filter(lambda x: not x.is_alive(), threads))) != len(threads): pass

for t in tasknoorder : #this loop is used to extract tasks from task who are still not excuted and their

    # verify if all the dependencies of the task t are in the list of the executed tasks(tasksexecuted)
    #if it is the cas we add it to tasks to be executed
    if all(elem in list(map(lambda x: x.name, taskexcuted)) for elem in self.getDependancies(t.name)):
        tasktobexcuted.append(t)
```

SEQUENTIEL VS PARALLELISME

Exécution des tâches en séquentiel

```
list1 = TaskSystem([t1, t2, t3, t4, t5],
                   {'T1': [], 'T2': ["T1"], 'T3': ["T1", "T2"], 'T4': ["T1", "T2"], 'T5': []})


start= time.perf_counter()
runT1()
runT5()
runT2()
runT3()
runT4()
finish= time.perf_counter()
print(f'Finished in {round(finish-start, 2)} seconds(s)')
```

```
PS C:\Users\Feriel Touati> & "C:/Users/Feriel Touati/AppData/Local/Microsoft/Windows
this is task 1 : 1
this is task 5
this is task 2 : 2
this is task 3 : 3
this is task 4 : -1
Finished in 18.05 seconds(s)
PS C:\Users\Feriel Touati>
```

Temps d'exécution :

18.05 Secondes

SEQUENTIEL VS PARALLELISME

Exécution des tâches avec la parallélisation

```
list1 = TaskSystem([t1, t2, t3, t4, t5],
                   {'T1': [], 'T2': ["T1"], 'T3': ["T1", "T2"], 'T4': ["T1", "T2"], 'T5': []})

start= time.perf_counter()
list1.Run()
finish= time.perf_counter()
print(f'Finished in {round(finish-start, 2)} seconds(s)')
```

```
PS C:\Users\Feriel Touati> & "C:/Users/Feriel Touati/AppData/Local/Microsoft/WindowsApp
this is task 1 : 1
this is task 5
this is task 2 : 2
this is task 3 : 3
this is task 4 : -1
Finished in 10.12 seconds(s)
PS C:\Users\Feriel Touati>
```

Temps d'exécution :

10.12 Secondes

MERCI
DE
VOTRE
ATTENTION