Documentation utilisateur

Bruce MILPIED, Bowen ZHANG, Nicolas SUDRES, Ronan KOMPF

Sommaire

Récupération du programme

Depuis GitHub

Pour récupérer le programme depuis GitHub, vous avez deux solutions :

- Soit par ce lien. Il faut ensuite cliquer sur `code`, puis `Download ZIP`.
- Soit par commande dans le terminal avec la commande suivante :

```
wget https://github.com/d3vyce/temp_reel
```

Depuis BlackBoard ESIEE

Pour télécharger le fichier depuis BlackBoard `Mes groupes`, puis dans `Groupe 3 E`.

Décompression du fichier

Après avoir téléchargé le fichier temp_reel_groupe_3E.tgz, pour le désarchiver, tapez la commande suivante :

```
dpkg -i temp_reel_groupe_3E.tgz
```

Cela devrait créer l'arborescence suivante :

```
temp_reel_groupe_3E/
        |-- doc/
        |   |-- doc_user.pdf
        |   `-- doc_dev.pdf
        |-- exemple/
        |   |-- test1
        |   `-- test2
        |-- part1/
        |   |-- lib/
        |   |   |-- algo.c
        |   |   |-- algo.h
        |   |   |-- part1.c
        |   |   |-- part1.h
        |   |   |-- sorted_job_list.c
        |   |   `-- sorted_job_list.h
        |   |-- graph.py
        |   `-- makefile
        |-- part2/
        |   |-- lib/
        |   |   |-- worst_case_fp.c
        |   |   |-- worst_case_fp.h
        |   |   |-- part2.c
        |   |   `-- part2.h
        |   `-- makefile
        `-- README
```

Installation

Python

Pour installer Python, utiliser la commande suivante :

```
sudo apt install python3
```

Vous pouvez ensuite vérifier l'installation de Python avec la commande `python3` :

```
Python 3.9.2 (default, Mar 15 2021, 17:37:51)
[Clang 12.0.0 (clang-1200.0.32.29)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
```

Xlsxwriter

Cette extension est utilisée dans le fichier `graph.py`, elle permet de générer des fichiers Excel. Pour l'installer, utiliser la commande suivante :

```
pip install --user xlsxwriter
```

Compilation des parties 1 et 2 du programme

Pour compiler les parties 1 et 2, il faut utiliser les commandes suivantes :

```
cd temp_reel_groupe_3E/
cd part1 && make && make clean && cd ../
cd part2 && make && make clean && cd ../
```

Si tout s'est bien déroulé, un fichier exécutable `part1` est apparu dans le dossier `/Part1` et un fichier exécutable `part2` est apparu dans le dossier `/part2`.

Utilisation Part 1 : Visualisation d'un chronogramme

Programme

Pour éxecuter le programme, il faut utiliser la commande suivante :

```
./part1/part1 [File] [Algorithme] [Duree]
```

File

Les fichiers utilisés par le programme sont situés dans le dossier exemple. Les fichiers sont constitués de deux éléments :

```
[nombre de tache]
[Cn] [Dn] [Tn]
...
```

⚠️ Le nombre de tâches doit être égal au nombre des lignes suivantes.
Ex : s'il y a 3 tâches, il doit y avoir 3 lignes en dessous avec la structure `[Cn] [Dn] [Tn]`.

exemple avec le fichier `test2` :

```
3
2 5 7
3 7 11
5 10 13
```

Algorithme

Deux algorithmes sont disponibles :

- `FP` (fixed priorities) : les priorités sont définies par l'ordre donné dans le fichier.
- `EDF` (dynamic priorities) : les priorités sont données par les échéances absolues des tâches.

Durée

Correspond au nombre d'unités de temps pour lequel le programme doit s'exécuter.

Exemple d'une commande d'exécution

`./part1/part1 exemple/test2 edf 30`. Exécute l'algoritme edf avec le fichier test2 pendant 30 unités de temps.
L'output de la commande devrait être le suivant :

```
 Algo EDF :
1 1 2 2 2 3 3 3 3 3 1 1 2 2 2 1 1 3 3 3 3 3 1 1 2 2 2 3 1 1
```

Un fichier `output` est créé dans le même temps, ce fichier sera utilisé dans la partie suivante pour l'affichage graphique.

Affichage graphique du résultat

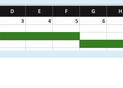
⚠️ Avant de faire l'affichage graphique, il faut exécuter l'algorime et vérifier qu'un fichier `output` a bien été généré.

Pour générer l'affichage graphique, il faut utiliser la commande suivante :

```
python3 graph.py
```

Cela devrait générer le fichier `graph.xlsx` qui représente de manière graphique l'output du programme.

Si on prend le fichier `output` de l'exemple précédent, le fichier Excel devrait ressembler à cela :



Utilisation Part 2

Programme

Pour exécuter le programme, il faut utiliser la commande suivante :

```
./part2/part2 [File]
```

File

Les fichiers ont la même structure que pour la partie 1. Des fichiers exemple sont dans le dossier `exemple`.

Exemple d'une commande d'exécution

`./part2/part2 exemple/test2`.

```
Résultat de la fonction test_load : 1

Tache 1 :
Résultat de la fonction get_busy_period : 2
Nombre d'instances pendant la busy period  : 1
Pire temps de réponse : 2

Tache 2 :
Résultat de la fonction get_busy_period : 5
Nombre d'instances pendant la busy period  : 1
Pire temps de réponse : 5

Tache 3 :
Résultat de la fonction get_busy_period : 39
Nombre d'instances pendant la busy period  : 3
Pire temps de réponse : 17
```

Bug report

Pas de bug