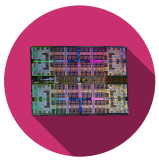


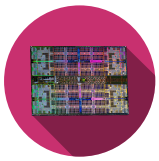
Analyse de Programmes Mystères

Labor Architecture des ordinateurs

Contenu

1 Objectifs

L'objectif de ce mini-labo est d'analyser des programmes inconnus avec des outils de débogage tels que *Activity Monitor* (MacOs), *Task Manager* (Windows) ou *btop* (Linux, MacOS, Windows) ainsi que des outils d'analyse de performance tels que **hyperfine** (Linux, MacOS, Windows) ou **time** (Linux, MacOS).



2 Installation

Tout d'abord, vous devez installer les différents outils que nous utiliserons pour les tests de performance.

2.1 `hyperfine` et `time`

`hyperfine` est une application de benchmarking en ligne de commande.

Pour installer **`hyperfine`** lisez la description dans le **`README`** du dépôt https://github.com/sharkdp/hyperfine?tab=readme-ov-file#installation.

Le programme **`time`** est déjà installé sur MacOS et Linux, il n'est pas disponible sur Windows.

2.2 `btop`

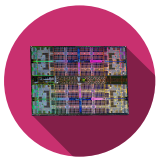
`btop` est un outil de surveillance du système basé sur le terminal similaire aux outils intégrés tels que *Task Manager* (Windows) ou *Activity Monitor* (MacOS).

Pour installer **`btop`** lisez la description dans le **`README`** pour MacOS et Linux lisez la description dans le dépôt officiel https://github.com/aristocratos/btop?tab=readme-ov-file#installation pour Windows utilisez le fork **`btop4win`** https://github.com/aristocratos/btop4win?tab=readme-ov-file#installation.

2.3 Verifier l'installation

Pour vérifier que l'installation a réussi, exécutez les commandes suivantes dans un terminal :

```
hyperfine --version
btop      --version
time time           # Linux MacOS only
```



3 Analyse des programmes mystères

Les programmes binaires dans le dossier **`car_labs/dbg/rust-mystery/release/`** peuvent être exécutés avec différents paramètres.

```
Usage: rust_mystery_v1_0_0_Mac_AARCH64 [OPTIONS]

Options:
  -m, --mystery <MYSTERY>
  -h, --help                 Print help
  -V, --version              Print version
```

Selon le système d'exploitation, un autre fichier binaire doit être exécuté.

- **`rust_mystery_v1_0_0_Mac_AARCH64`** pour MacOS
- **`rust_mystery_v1_0_0_Linux_x64`** pour Linux
- **`rust_mystery_v1_0_0_Windows_x64.exe`** pour Windows

Adaptez en conséquence les commandes ci-dessous.

L'option **`-m`** ou **`--mystery`** attend une valeur de **`1`** à **`5`**. Chaque valeur entraîne un comportement différent du programme.

Le programme peut être exécuté directement avec l'option **`-m`**.

```
./rust_mystery_v1_0_0_Mac_AARCH64 -m 1
```

Les programmes ne durent que très peu de temps. Pour mesurer la vitesse d'exécution, nous utilisons l'outil **`hyperfine`** ainsi que **`btop`**.

Commencez par démarrer **`btop`** dans un terminal séparé pour surveiller l'utilisation du système. Ensuite, vous pouvez exécuter le programme respectif avec **`hyperfine`**. Par exemple, Mystery 4 :

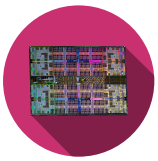
```
hyperfine --warmup 3 --export-markdown mystery-4.md --show-output --min-runs 10
"release/rust_mystery_v1_0_0_Mac_AARCH64 -m 4 -v"
```

Exécutez toutes les variantes **`-m 1`** à **`-m 5`** du programme et analysez la sortie de **`hyperfine`** et **`btop`**.

3.1 Analyse avec `hyperfine`

À la fin d'un benchmark **`hyperfine`**, un résumé de la vitesse d'exécution est affiché.

```
Time (mean ± σ):      64.9 ms ±  13.3 ms    [User: 14.8 ms, System: 12.5 ms]
Range (min … max):    58.3 ms … 147.6 ms    43 runs
```



Dans ce cas, le programme a été exécuté 43 fois. En moyenne une exécution a duré **`64.9ms`** avec une variation de **`13.3ms`**. La durée d'exécution minimale était de **`58.3ms`** et la maximale de **`147.6ms`**.

Les valeurs **`User`** et **`System`** sont également importantes. Ces valeurs indiquent combien de temps le programme a passé dans l'espace utilisateur et système. Dans ce cas, **`14.8ms`** et **`12.5ms`**.

Regardez le temps d'exécution, les valeurs utilisateur et système pour tous les programmes. Essayez de comprendre pourquoi les programmes prennent des durées différentes.

Notez vos conclusions. Quelles opérations ces programmes effectuent-ils probablement?