

TP1 – Systèmes d’exploitation (Windows, MacOs, Linux)

Par  
Frédérik Boutin

Travail présenté à Mme Lise Boudreault

Dans le cadre du cours Systèmes d’exploitationINF34207-MS

14 février 2024

# Introduction

## 1.1 Dans un tableau, présentez les caractéristiques (lignes) de l’ordinateur que vous utilisez pour ce cours :

Avant-propos : Il est à noter que pour ce travail, j’ai utilisé mon ordinateur portable. Lors de la séance Zoom, j’avais inscrit dans le chat les caractéristiques de mon pc de type tour de bureau. Ainsi, il est normal qu’il ne s’agisse pas des même caractéristiques.

|  |  |
| --- | --- |
| **Caractéristique de l’ordinateur** | **Valeur de la caractéristique** |
| Nom du processeur | 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz, 2803 Mhz, 4 Core(s), 8 Logical Processor(s) |
| Manufacturier du processeur | Intel Corporation |
| Vitesse de base du processeur | 2.80GHz |
| Nombre de cœur du processeur | 4 |
| Taille de la cache de niveau 1 | 320 KB |
| Taille de la cache de niveau 2 | 5.0 MB |
| Taille de la cache de niveau 3 | 12 MB |
| Capacité de la mémoire vive | 16.0 GB |
| Vitesse de la mémoire vive | 3200 MHz |
| Type de mémoire vive | DDR4 |
| Nom du disque dur principal (modèle) | OS (C :) |
| Manufacturier du disque dur principal | (Standard disk drives) |
| Capacité du disque dur principal | 475 GB |
| Système de fichiers utilisé du disque dur principal | NTFS (New Technology File System) |
| Type de connexion du disque dur principal (SATA, USB, NVMe, etc.) | NVMe INTEL SSDPEKNW512G8 |
| Nom de la carte graphique (si présente) | Mon ordinateur n’a pas de carte graphique, cependant il a une solution graphique intégrée, soit Intel Iris Xe Graphics. Cela signifie que l’unité de traitement graphique (GPU) est intégrée directement dans la puce du processeur, partageant des ressources telles que la mémoire avec le CPU. |
| Manufacturier de la carte graphique | Intel Corporation (manufacturier de la solution graphique intégrée ≠ carte graphique) |
| VRAM de la carte graphique | Intel Iris Xe Graphics dispose de VRAM, mais il ne s’agit pas d’une mémoire physique séparée comme pour les cartes graphiques. Au lieu de cela, elle utilise une partie de la mémoire principale (RAM) du système comme mémoire vidéo partagée. |
| Vitesse de l’horloge de la carte graphique | Après avoir essayé sans succès de trouver ce renseignement utilisant Task Manager et Intel Graphics Command Center, je n’ai pas trouvé. Il faudrait utiliser un logiciel tiers, mais je ne préfère pas m’y risquer. |

Les outils que j’ai utilisés pour trouver les différentes informations sont les suivant :

* System Information
* Task Manager
* Device Manager
* Windows Power Shell en mode admin (commande Get-CimInstance -ClassName Win32\_PhysicalMemory | Format-Table SMBIOSMemoryType pour obtenir le type de mémoire vive)

## 1.2 Lorsque vous utilisez votre ordinateur, décrivez au moins deux applications (p.ex. un jeu vidéo, un outil de montage, un outil de modélisation 3D, etc.) :

La première application est Google Chrome (version : 120.0.6099.225) qui est mon navigateur web. Cette application est plus efficiente en termes de temps de réponse, puisque les navigateurs modernes sont conçus pour optimiser leur temps de réponse en fonction des ressources matérielles disponibles, telles que la mémoire RAM et les processeurs multicœurs. Parmi les exemples de moyens d’optimisation, il y a l’architecture multiprocessus qui permet une meilleure isolation entre les différents processus de l’application et une meilleure répartition des tâches sur plusieurs processus (temps de réponse plus rapide), l’utilisation d’une programmation asynchrone pour la gestion des tâches, une gestion intelligente des ressources permettant l’attribution de priorité aux onglets actifs et l’utilisation du chargement différer pour les ressources non prioritaire, ainsi que l’utilisation d’un moteur de rendu optimisé qui utilise le patron de conception Composite afin de permettre la gestion des pages par arborescences d’objet, ce qui améliore donc le temps d’opération sur les éléments des pages. Durant un test d’utilisation pour la lecture d’une vidéo YouTube (musique), l’application utilisait 2-5% du CPU, 650-750 MB de RAM, 0-0.1% du disque et 0-0.4% du réseau et la lecture était fluide.

La seconde application est mon logiciel gratuit d’antivirus AVG Internet Security System (version 23.12.8700.0). Cette application, lorsqu’elle effectue une analyse du système, elle doit accéder à plusieurs fichiers du système et effectuer des opérations de comparaison avec sa source de données afin de détecter les menaces. Ainsi, tout ce processus occasionne une grande utilisation des ressources matérielles de l’ordinateur, autant au niveau des calculs du processeur, de la mémoire vive utilisé par l’application pour ses comparaisons et au niveau du disque pour l’accès aux fichiers. On sait que l’accès au disque est la forme d’opération la plus lente sur une machine et l’application doit y accéder souvent pour l’analyse, ainsi il est possible de comprendre pourquoi elle possède un temps d’exécution plus lente. Un test d’analyse intelligente a occasionné des utilisations de 20-70% du CPU, 150-300 MB de RAM, 1.5-21% du disque.

En résumé, comme il a été possible de le voir à travers ces deux applications, la différence entre l’efficience d’exécution d’une application va beaucoup dépendre des moyens et des principes de conception qu’elle utilise pour utiliser au mieux les ressources matériels de la machine, le type de ressource qu’elle doit utiliser sur la machine, ainsi que la possibilité ou non d’optimiser leurs processus. Ainsi, un navigateur web qui effectue une bonne séparation de tâches peut les exécuter plus rapidement et en parallèle, qui utilise une cache pour certaines informations y aura accès plus rapidement, qui utilise le chargement différer permettant le rendu de page rapide pendant que le reste des éléments secondaires continue de se générer. À l’opposé, l’application antivirus doit nécessairement faire son analyse de fichiers en passant par le disque, ce qui augmente le temps de traitement.

# Programmation

## 2.1 Choisissez trois langages de programmation (ex: C, C++, C#, JavaScript, Java, Python) :

J’ai choisi les langages Python, C# et C++. Les images ci-dessous présentent les trois codes de ces langages. Les trois programmes suivent la même logique, c’est-à-dire qu’ils font des appels de fonctions que j’ai créé pour représenter la saisie d’information, l’affichage à l’écran et l’impression à l’imprimante. Pour chacun des cas, la fonction de saisie ce nomme EnteringInformation, la fonction d’affichage à l’écran ce nomme DisplayInformation et finalement, la fonction se nomme Printing.

## 2.2 Présentez le code requis pour effectuer trois requêtes vers le système d’exploitation :

### Saisie d’une information (entrée de données au clavier)



Figure 1 - Code Python pour une saisie



Figure 2 -Code C# pour une saisie



Figure 3 -Code C++ pour la saisie

### Affichage de l’information à l’écran



Figure 4 -Code Python pour l'affichage



Figure 5 -Code C# pour l'affichage



Figure 6 -Code C++ pour l'affichage

### Impression de la même information qu’à l’étape précédente, mais à partir d’une imprimante, donc, effectuer un appel à l’imprimante (par défaut)



Figure 7 -Code python pour l'impression



Figure 8 -Code C# pour l'impression





Figure 9 -Code C++ pour l'impression

## 2.3 Au niveau de la complexité de la programmation, que remarquez-vous entre ces trois langages ?

Il est possible de remarquer que plus le langage de programmation est de bas niveau, ou dans autrement dit, plus le langage est près des instructions machines, plus la programmation est complexe. Il est d’ailleurs possible de le voir entre les trois segments des codes, particulièrement pour l’exemple de l’impression. En effet, en examinant les figures 7 à 9, nous remarqu’on que le code Python nécessite X lignes de code, alors que les codes équivalents en C# et C++ ont respectivement nécessité Y et Z lignes de code. Ainsi, même si chacun des langage utilise la même bibliothèque (Window.h) pour les fonction de l’imprimente par défaut, la comparaison démontre bien que le langage de plus haut niveau qu’est Python effectu un grand nombre d’opération supplémentaire en arrière-plan par rapport aux deux autres langages de plus bas niveau. Puisque c’est le système d’exploitation qui s’occupe de

**(3)** **Bases** de données 5%

* **3.1** Quel est le rôle du système d’exploitation (séquence d’étapes) lorsqu’un langage de programmation transmet une requête vers un serveur de base de données (ex : MySQL)?

**NOTE**: Attention à la qualité du français, perte maximale de 10% soit -2,5%