**Architecture Project**

**Group Members:** Martin Navarrete, CJ Faircloth, Danny Pham, Michael Canales,  Thomas Medina-Herrera

**Introduction to the Project:**

Our project aims to explore sorting algorithms and how they perform on various hardware setups. We picked this project because understanding algorithm efficiency across different platforms helps us optimize performance and identify potential resource bottlenecks.

**Background:**

**Systems:**

1) ThinkPad T440p:

- OS: EndeavourOS Linux x86\_64

- CPU: Intel i7-4702MQ

- Memory: 15875 MB

2) Asus 14X:

- OS: Windows 11

- CPU: Intel i7-13700H

- Memory: 15600 MB

3) Dell Inspiron 16:

- OS: Windows 11

- CPU: 13th Gen Intel(R) Core (TM) i7-1360P

- Memory: 16384 MB

4) Dell G15 Gaming Laptop:

- OS: Windows 11

- CPU: AMD Ryzen 7 5800H

- Memory: 8000 MB

5) Dell Inspiron 15:

- OS: Windows 10

- CPU: Intel i5-7300HQ

- Memory: 8000 MB

**Procedure:**

**What we did:** Our plan involved running a C++ program designed to sort identical arrays using a variety of sorting algorithms, while comparing the sorting times across two distinct hardware setups.

**How we did it:** We crafted a C++ program featuring an array with a user-defined size of 100,000 in our case, comprising of a variety of 60 random characters. This program was implemented using algorithms including selection sort, insertion sort, quick sort, and count sort. Additionally, our program meticulously tracked CPU completion time for each algorithm. To ensure consistency, we employed a seeding mechanism for generating random characters, thus replicating the same array for both hardware configurations under scrutiny.

**Analysis of results:**

**Data:** Below is the average runtime of each algorithm for each device

ThinkPad:

* Selection Sort: 341.167 sec
* Insertion Sort: 178.825 sec
* Count Sort: 0.004384 sec
* Quick Sort: 0.049913 sec

Asus Zenbook 14X:

* Selection Sort: 83.6922 sec
* Insertion Sort: 75.595 sec
* Count Sort: 0.002 sec
* Quick Sort: 0.0139 sec

Dell Inspiron 16:

* Selection Sort: 181.013 sec
* Insertion Sort: 0.002031 sec
* Count Sort: 0.003743 sec
* Quick Sort: 0.032 sec

Dell G15 Gaming Laptop:

* Selection Sort: 118.8932 sec
* Insertion Sort: 113.9474 sec
* Count Sort: 0.0034 sec
* Quick Sort: 0.030263 sec

Dell Inspiron 16:

* Selection Sort: 181.013 sec
* Insertion Sort: 0.002031 sec
* Count Sort: 0.003743 sec
* Quick Sort: 0.0139 sec

**Analysis**:

* On average the best algorithm was Count Sort, its time complexity is O (n + k). It is effective because the range of input values (60) compared to the number of elements to be sorted (500000).
* Average the worst algorithm was Selection Sort, its time complexity is O(n^2). Its performance degrades significantly as the size of the dataset increases.

**Conclusion:**

Results:

* Across the board the Dell Inspiron 5 ran the worst, probably because of the older CPU model than the other computers.
* Count sort is more efficient when dealing with a small variety of possible values. In our case, characters 'A' to '}'; A range of 60 possible values.
* Which system allowed the algorithms to perform the best?
  + Asus Zenbook 14X; Due to more recent architecture and performance-based CPU.
  + e.g. i7-13700H with the 'H' series being a more "high performance" variant of their CPU's.