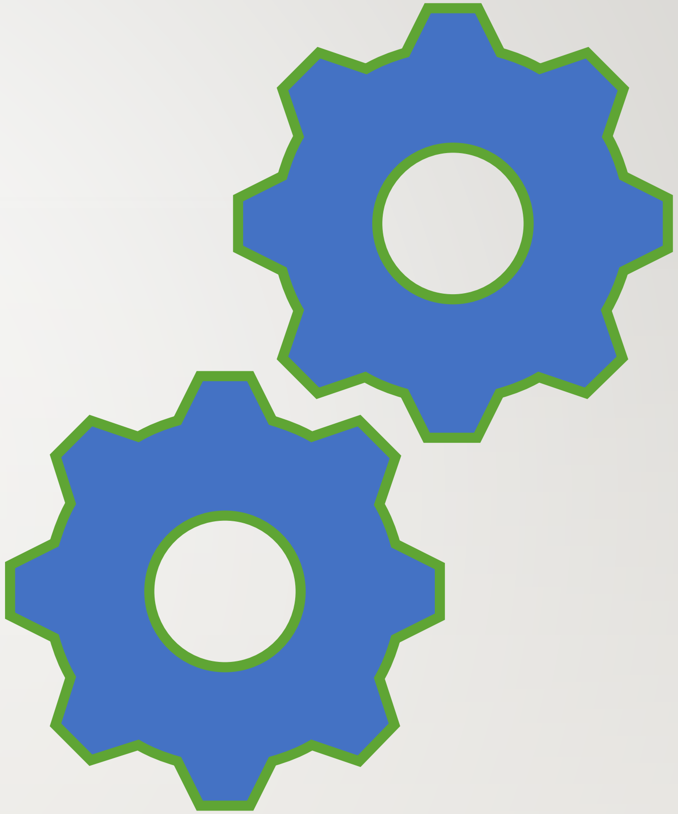
Algorithm Engineering Project



TEAM 1

MERT SANER- M.Sc. SCIENTIFIC COMPUTING

MAX PAUL BÄHNISCH - M.Sc. COMPUTER SCIENCE

HANNAH WETTERAU - M.Sc. COMPUTER SCIENCE

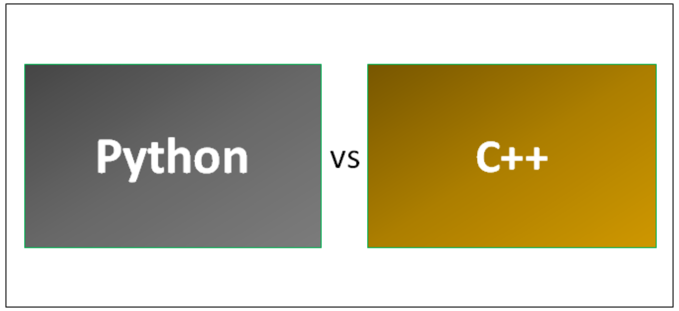
Programming Language Choice

- **In this Algorithm Engineering project, C++ programming language is used.**
- C++ is high-level, high-performance, object-oriented programming language.
- **Main advantages of C++,** portability, scalability, multi-paradigm programming, low-level manipulation, and large community
- C++ also provided more control over hardware aspects like memory management and CPU usage.

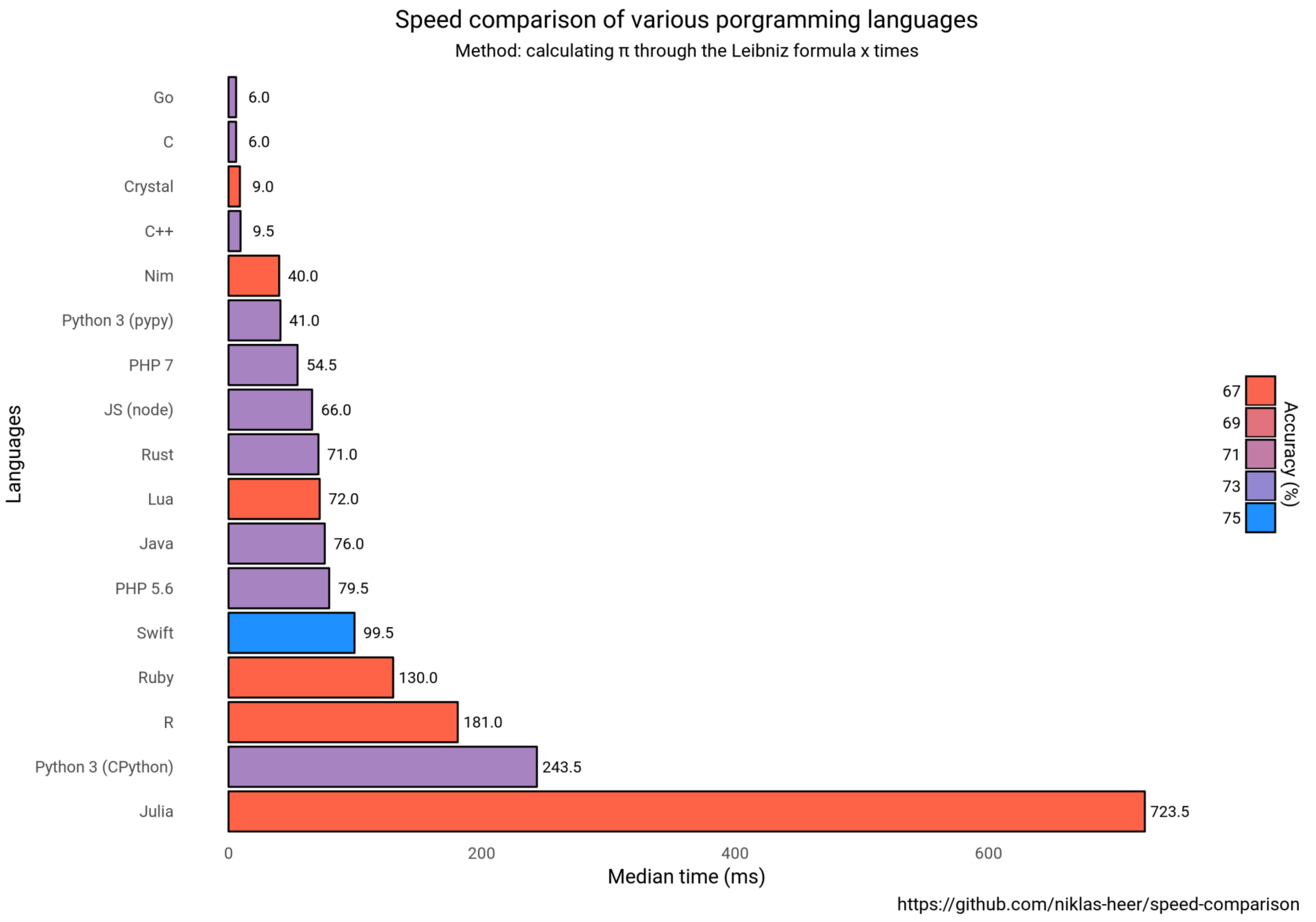
Crucial Difference of C++

- The most important reason to choose C++ was the **speed**.
- In real-world and real applications it is faster than java
- Thanks to **lighter memory footprint that results in better cache performance**.
- Particularly, for sorting and processing the data, **<u>C++ is noticeably faster.</u>**

Comparison with Python



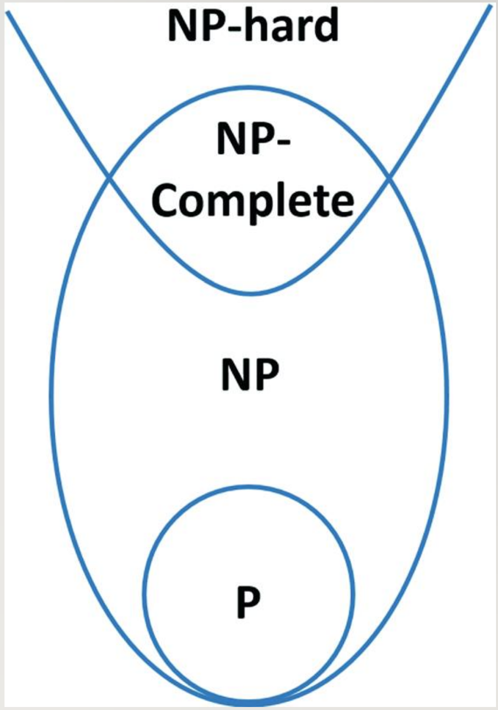
- Python is **interpreted**, while C++ is **compiled**.
- Compiled languages are converted directly into machine code that the processor can execute.
- Interpreted code is always slower than direct machine code because it takes a lot more instructions to implement an interpreted instruction than to implement an actual machine instruction.
- **As a result, compiled languages tend to be faster and more efficient to execute than interpreted languages.**



Speed comparison of various porgramming languages
Method: calculating π through the Leibniz formula x times
Languages
Go
6.0
C
6.0
Crystal
9.0
C++
9.5
Nim
40.0
Python 3 (pypy)
41.0
PHP 7
54.5
JS (node)
66.0
Rust
71.0
Lua
72.0
Java
76.0
PHP 5.6
79.5
Swift
99.5
Ruby
130.0
R
181.0
Python 3 (CPython)
243.5
Julia
723.5
0
200
400
600
Median time (ms)
Accuracy (%)
67
69
71
73
75
https://github.com/niklas-heer/speed-comparison

Weighted Cluster Editing Problem

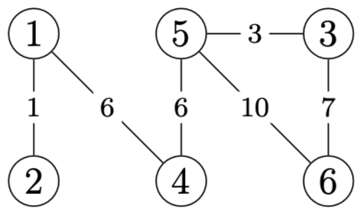
- The NP-hard Cluster Editing problem is among the best-studied parameterized problems.
- The Cluster Editing problem is defined in a way that given an undirected, loopless graph, it is needed to find a set of edge modifications (insertions and deletions) of minimum cardinality, such that the modified graph consists of disjoint cliques.
- An undirected graph G= (V, E) and a non-negative integer, the NP-hard Cluster Editing problem asks whether can be transformed into a disjoint union of cliques by modifying at most edges.



Implementation Highlights

- We tried two approaches for implementation
- **Naïve Approach and Recursive Approach**

Input graph:



Input file:

```
6
1 2 1
1 3 -5
1 4 6
1 5 -1
1 6 -4
2 3 -4
2 4 -2
2 5 -10
2 6 -8
3 4 -5
3 5 3
3 6 7
4 5 6
4 6 -4
5 6 10
```

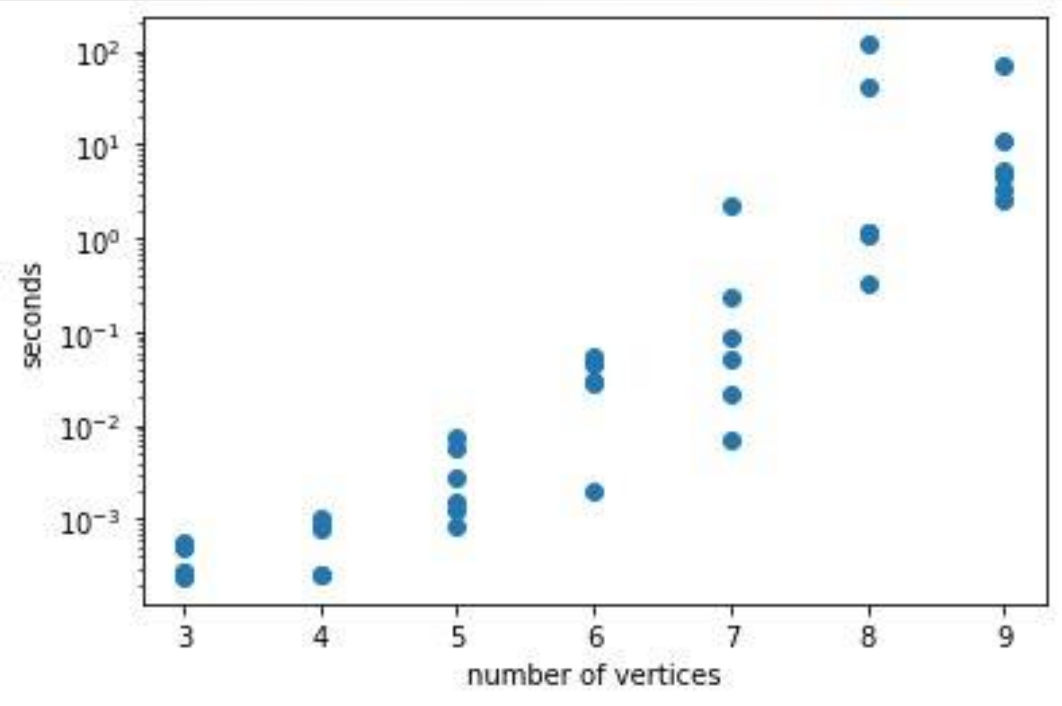
Experiments

- While implementing our project we created adjacent matrices and counted for a number of vertices and clusters. We also calculated the cost of each such change.
- We answered one of the important questions of how does the running time depend on the number of vertices of the input graph, the answer is, for vertexcount vertices, we need to test 2^((((vertexcount - 1)^2)(vertexcount-1)) / 2) computations.
- For usual computers, it would be too time-consuming to calculate.

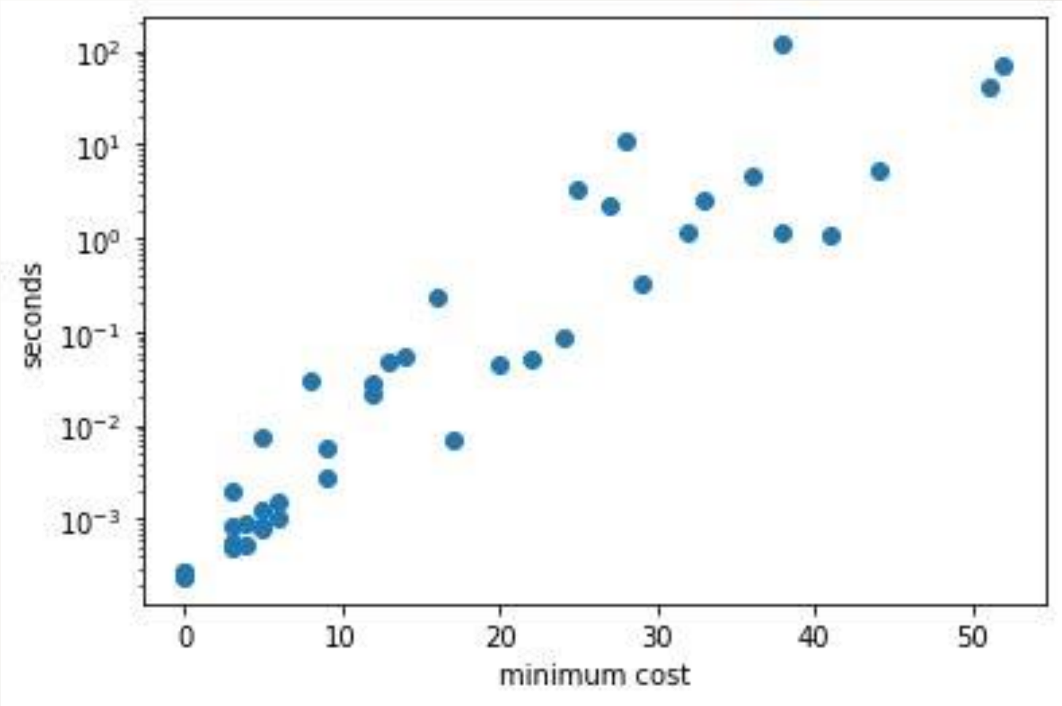
Final Remarks

- With the naive approach and recursive approach, the running time increases with the cost.
- With each cost, we need to build a new recursive tree with one additional layer, while it depends on the cost of the vertex change.
- It can be stated that our solver worked correctly to perform calculations of the cost.

Graphs



seconds
10^2
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
3
4
5
6
7
8
9
number of vertices



seconds
10^2
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
0
10
20
30
40
50
minimum cost