Ex04_p02

- To calculate the matrix products i wrote a function **matrix_mult**. The function simply calculates the matrix product of two matrices and return the CPU time spent.

p02.cpp

```
double matrix_mult(std::vector<std::vector<int>> &a,
std::vector<std::vector<int>> &b)
{
    std::vector<std::vector<int>> c(N, std::vector<int>(N));

    auto start = std::chrono::high_resolution_clock::now();

    for (int i = 0; i < N; i++) {
        for (int j = 0; j < N; j++) {
            for (int k = 0; k < N; k++) {
                c[i][j] += a[i][k] * b[k][j];
            }
        }
    }

    auto finnish =
std::chrono::high_resolution_clock::now();
    std::chrono::duration<double> elapsed = finnish - start;

    std::vector<std::vector<int>>().swap(c);

    return elapsed.count();
}
```

- To measure CPU time I ran the program n = 50 times and then calculated the average using a python script

scrip.sh

```
for (( i=1; i<=$n; i++ ))
do
    echo "Iteration ${i}"
    ./$executable_1 >> $out1
    ./$executable_1 "b" >> $out2
    ./$executable_1 "c" >> $out3
    ./$executable_2 >> $out4
    ./$executable_2 "b" >> $out5
    ./$executable_2 "c" >> $out6
done
```

medians.py

```
import numpy as np
import sys

for file in sys.argv[1:]:
    with open(file, "r") as a_out:
        lines = a_out.readlines()
        values = np.array([float(line.strip()) for line in
lines])
        print(f'Average of {file.replace("_output.txt",
"")}: {values.mean()} s')
```

Results:

- As expected, when running the program with -O3 optimisation, the performance was significantly better. Also the difference in performance between (a), (b) and (c) was very small, however, there is a very small bump in performance from (a) to (b) to (c).

output from medians.py

```
Average of out/p02_O0_a_output: 5.053605800000001 s
Average of out/p02_O0_b_output: 5.0517858 s
Average of out/p02_O0_c_output: 5.018779 s
Average of out/p02_O3_a_output: 0.9636708999999999 s
Average of out/p02_O3_b_output: 0.9638317399999999 s
Average of out/p02_O3_c_output: 0.9384324 s
```