Profiling protocol for IVS project

1. Introduction

This document shows the results of profiling our math library with the goal of finding the weak spot in our code to optimize

2. Testing

For profiling purposes we implemented a function to calculate the selected standard deviation.

We had three testing sets with the number of numbers of 10, 100 and 1000. As our main profiling approach we used code instrumentation.

3. Conclusion

The data shows, that the most called functions were addUp (basic addition) and toPower. In case of N input numbers, the addUp function was called N * 2 times, and toPower was called N + 1 times.

In case of data entry of 10 numbers, the program spent most of the time in other functions than those two i.e . squareRoot, multiply, subtract and divide. In case of 100 numbers, the program time between these two sections was very similar and in case of 1000 numbers, most of the program time was spent on function addUp and toPower.

It is logical to assume that the optimalization course is to try to improve the addUp and toPower functions as they are called the most, but all of our functions are from the standard javascript Math library, and they are very well optimized, so from the way we see it, there is small room for improvement.