

Task 4

Durch das Skript perf_analysis.sh erhielten wir folgende Messdaten in input.res :

```
DATA 1204.566872 1109.259352 1059.466358 1151.623018 997.861061 1130.606710
1051.786520 1207.386171 1069.902174 1050.898945
DATA 1595.373213 1622.197985 1607.928089 1557.232466 1637.642473 1580.457531
1610.193979 1646.972878 1594.052422 1569.531898
DATA 2145.281046 2108.344366 2214.109409 2135.561724 2094.977406 2154.362816
2164.870365 2130.281111 2286.721819 2146.860229
DATA 2721.511223 2660.196569 2777.731150 2749.738516 2919.476835 3088.050816
2762.431318 2679.066442 2835.972158 2927.159334
DATA 3308.794941 3363.542452 3286.505058 3292.140289 3275.766235 3417.373285
3395.399517 3228.238573 3437.475658 3230.716019
```

Die weitere Analyse mit extra-p ergab als Modell $(577.649) + (1.13338e-05 * x^{(1)}) * (\log_2(x))^{(1)}$). Die Laufzeit unseres Algorithmus hat also eine Komplexität von $k+l*x*\log_2(x)^1$, wobei l und k Konstanten sind.

Dieses Laufzeit Modell weist große Ähnlichkeiten mit dem in Aufgabe 3 ermittelten auf. Dort kamen wir auf eine Komplexität von $3n+n*\log(n)$. $3n$ würde also gerade $k+l$ entsprechen, was bei einem großen n ein Vorteil wäre.