

# Määrittelydokumentti

Joel Järvinen

February 19, 2014

## 1 Ongelma / Aihe

Tietorakenteiden ja algoritmien harjoitustyönä tulen tekemään erilaisia järjestämisalgoritmeja ja vertailemaan niitä keskenään, tulen myös vertailemaan omia järjestysalgoritmeja Javan Arrays.sort()-metodiin. Tietorakenteena tulen toteuttamaan geneerisen keon.

## 2 Tavoitevaatimukset

### 2.1 Kekojärjestäminen - Heap Sort

Aikavaativuus kekojärjestämiselle on  $\mathcal{O}(n \log n)$ , ja kekojärjestämisessä käytetyn heapify-operaation rekursioiden vuoksi tilavaativuus on  $\mathcal{O}(\log n)$ .

### 2.2 Kuplajärjestäminen - Bubble Sort

Aikavaativuus kuplajärjestämisellä on  $\mathcal{O}(n^2)$ , mutta tilavaativuus on  $\mathcal{O}(1)$  sillä algoritmi käyttää yhtä apumuuttujaa.

### 2.3 Lomitusjärjestäminen - Merge Sort

Aikavaativuus on  $\mathcal{O}(n \log n)$  ja tilavaativuus on  $\mathcal{O}(n)$ .

### 2.4 Pikajärjestäminen - Quick Sort

Aikavaativuus pikajärjestämisellä on pahimmassa tapauksessa  $\mathcal{O}(n^2)$ , mutta keskimääräinen aikavaativuus on  $\mathcal{O}(n \log n)$ . Tilavaativuus on osittain iteratiivisesti tehdyllä pikajärjestämisellä  $\mathcal{O}(\log n)$ , mutta jos se tehdään kokonaan rekursiivisesti on tilavaativuus  $\mathcal{O}(n)$ .

### 2.5 Laskemisjärjestäminen - Counting Sort

Laskemisjärjestämisellä voidaan rikkoa  $\mathcal{O}(n \log n)$  aikavaativuusraja, mutta tällöin tulee tietää mikä on suurin mahdollinen luku mikä listalle on lisätty. Tällöin voidaan päästä aika- ja tilavaativuuksissa  $\mathcal{O}(n + k)$  luokkaan, missä  $n$  on listan koko ja  $k$  on suurin mahdollinen alkio.

### **3 Tavoitteet**

Pyrin kaikissa algoritmeissa saavuttamaan yllämainitut parhaat / keskimääräiset aika- ja tilavaativuudet.

## 4 Läheteet

Tietorakenteet ja algoritmit kurssin luentomateriaali  
StackOverflow sivusto *stackoverflow.com*