

## **Määrittelydokumentti**

### **Aihe**

Harjoitustyön tavoitteena on toteuttaa matriisilaskin, joka osaa suorittaa ainakin seuraavat toiminnot:

- Käyttäjän syöttämän matriisin tallentaminen ohjelmaan
- Kahden matriisin summan ja erotuksen laskeminen
- Matriisin kertominen vakiolla
- Matriisin kertominen toisella matriisilla
- Matriisin determinantin määrittäminen
- Käänteismatriisin määrittäminen
- Matriisin transpoosin määrittäminen

Ohjelman on myös osattava hoitaa tilanteet, joissa yritetään suorittaa matriisikertolaskua keskenään "erisuuruksille" matriiseille, tai halutaan selvittää käänteismatriisia sellaiselle matriisille joka ei ole kääntyvä.

Ohjelma saa syötteenä matriisit merkkijonona, komennot siitä mitä laskutoimituksia tehdään ja tiedot siitä mille matriiseille ne tehdään.

### **Algoritmit ja tietorakenteet**

Matriisien tallennus toteutetaan taulukkotietorakenteen avulla, sillä matriisin voi havainnollistaa helpoiten kaksiulotteisen taulukon avulla, jolle myös operaatioiden suoritus on kätevintä toteuttaa. Matriisikertolaskuihin sovelletaan joko standardia matriisikertolaskualgoritmia tai Strassen algoritmia riippuen siitä, kuinka haastavaksi Strassen algoritmi osoittautuu ja miten harjoitustyö edistyy. Projektin alkuvaiheessa toteutetaan tavallinen algoritmi, mutta tavoitteena olisi kuitenkin käyttää Strassen algoritmia, sillä se on tavallista algoritmia tehokkaampi. Determinaatin ja käänteisen matriisin määrittämiseen käytetään LU-hajoittelmaa, koska se on tehokkaimpia tapoja kyseisten ongelmien ratkaisemiseen.

### **Aika- ja tilavaativuudet**

Aikavaativuus on kertolaskun algoritmista riippuen joko  $O(n^3)$  tai  $O(n^{2.8074})$ .

### **Lähteet**

LU-hajotelma: [http://en.wikipedia.org/wiki/LU\\_decomposition](http://en.wikipedia.org/wiki/LU_decomposition)

Strassen algoritmi: [http://en.wikipedia.org/wiki/Strassen\\_algorithm](http://en.wikipedia.org/wiki/Strassen_algorithm)