Kauppamatkustajan ongelma

Sisällysluettelo

[Johdanto 1](#_Toc337890741)

[Testit 1](#_Toc337890742)

[Testien ajaminen 1](#_Toc337890743)

[Testien kattavuus 2](#_Toc337890744)

[Testien tulokset 2](#_Toc337890745)

[Kohteiden lukumäärä= 4 2](#_Toc337890746)

[Kohteiden lukumäärä= 6 3](#_Toc337890747)

[Kohteiden lukumäärä= 8 4](#_Toc337890748)

[Kohteiden lukumäärä= 10 4](#_Toc337890749)

[Kohteiden lukumäärä= 12 5](#_Toc337890750)

[Kohteiden lukumäärä= 16 5](#_Toc337890751)

[Kohteiden lukumäärä= 30,50,95 6](#_Toc337890752)

[Tarkka tulos vs Approksimaatio 7](#_Toc337890753)

# Johdanto

Tässä dokumentissa kuvataan Kauppamatkustajan ongelman ratkaisuun toteutetun Traveling Salesman –moduulin testauskattavuus ja –tulokset. Tuloksien analysoinnissa tutkitaan miten aikavaativuudet vastaavat vaadittuja.

## Testit

Moduulissa on kahdentyyppisiä testejä. Projektin algoritmeja ja tietorakenteita testaan suoraan ja niiden testauksessa pyritään testaamaan niiden oikeellisuutta. Näiden testien lisäksi testataan moduulin päärajapintaluokkaa TravelingSalesman erikokoisilla syötteillä suorituskyvyn mittaamiseksi. Testien kuvaukset löytyvät helpoiten JavaDoc:sta.

## Testien ajaminen

Testit on toteutettu JUnit-kehyksellä ja ne ajetaan Mavenin avulla joko

1. Komentoriviltä projektin juuresta komennolla **mvn test**tai
2. Netbeansin Projects-välilehdeltä klikkaamalla hiiren oikealla napilla projektia ja valitsemalla Test (tai painamalla Alt+F6 projektin ollessa auki)

## Testien kattavuus

Testien kattavuuden määritykseen on käytössä Clover-plugin. Kattavuusanalyysi saadaan ajamalla komentoriviltä projektin juuresta:

**mvn clover2:instrument clover2:aggregate clover2:clover**

Tämä luo kattavuusraportin hakemistoon:

**target/site/clover**

## Testien tulokset

Tässä kappaleessa keskitytään tulosanalyysissä vain suorituskykytesteihin. Suorituskykyä testattiin seuraavilla syötteillä:

|  |  |
| --- | --- |
| **Kohteiden lkm** | **Testattavat metodit** |
| **4** | Brute Force, Branch and Bound, Prim, Prim2, Greedy |
| **6** | Brute Force, Branch and Bound, Prim, Prim2, Greedy |
| **8** | Brute Force, Branch and Bound, Prim, Prim2, Greedy |
| **10** | Brute Force, Branch and Bound, Prim, Prim2, Greedy |
| **12** | Brute Force, Branch and Bound, Prim, Prim2, Greedy |
| **16** | Branch and Bound, Prim, Prim2, Greedy |
| **30** | Prim, Prim2, Greedy |
| **50** | Prim, Prim2, Greedy |
| **95** | Prim, Prim2, Greedy |

Jokaisen lukumäärän kohdalla testejä ajettiin 5 kertaa ja lopulliseen vertailuun laskettiin näiden testien keskiarvo. Etäisyydet solmujen välillä olivat kaikissa tapauksissa sattumanvaraisia lukuja välillä 1-100.

### Kohteiden lukumäärä= 4

Huom! Koska tämä testi ajettiin ensimmäisenä, niin siitä johtuen ajat ovat jonkin verran suurempia eivätkä siis vertailukelpoisia muiden testien kanssa.

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Brute Force | 0,396 | 0 |
| Branch And Bound | 0,292 | 0 |
| Prim(DFS) | 0,702 | 18.4 |
| Prim(Stack) | 0,556 | 34 |
| Greedy | 0,602 | 1.2 |

### Kohteiden lukumäärä= 6

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Brute Force | 0,01 | 0 |
| Branch And Bound | 0,01 | 0 |
| Prim(DFS) | 0,03 | 40.8 |
| Prim(Stack) | 0,01 | 33.6 |
| Greedy | 0,01 | 4.8 |

### Kohteiden lukumäärä= 8

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Brute Force | 0,84 | 0 |
| Branch And Bound | 0,37 | 0 |
| Prim(DFS) | 0,03 | 56,2 |
| Prim(Stack) | 0,03 | 63,4 |
| Greedy | 0,01 | 24,4 |

### Kohteiden lukumäärä= 10

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Brute Force | 68,36 | 0 |
| Branch And Bound | 4,48 | 0 |
| Prim(DFS) | 0,04 | 102,4 |
| Prim(Stack) | 0,014 | 161,6 |
| Greedy | 0,01 | 51,8 |

### Kohteiden lukumäärä= 12

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Brute Force | 6648,96 | 0 |
| Branch And Bound | 1682,18 | 0 |
| Prim(DFS) | 0,10 | 153,2 |
| Prim(Stack) | 0,14 | 190,4 |
| Greedy | 0,01 | 93,2 |

### Kohteiden lukumäärä= 16

|  |  |  |
| --- | --- | --- |
|  | **Aika** | **Matkaerotus  (tulos - optimi)** |
| Branch And Bound | 5632,73 | 0 |
| Prim(DFS) | 0,17 | 272,4 |
| Prim(Stack) | 0,06 | 264 |
| Greedy | 0,02 | 91,6 |

### Kohteiden lukumäärä= 30,50,95

Näillä kohdelukumäärillä ei voitu testata tarkkoja matkoja laskentatehon vuoksi, joten näissä mitattiin ainoastaan approksimaatioiden matkoja ja suorituskykyä. Alla olevissa taulukoissa tulokset:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Aika(50)** | **Aika(95)** | **Matka(50)** | **Matka(95)** |
| Greedy | 0,18 | 0,06 | 423,5 | 515 |
| Prim(DFS) | 1,38 | 0,44 | 1182,8 | 1983,4 |
| Prim(Stack) | 0,53 | 0,42 | 1732 | 3470,2 |

### Tarkka tulos vs Approksimaatio

Alla olevassa diagrammissa on vielä kuvattu approksimaatioiden suhdetta optimaalisen pituiseen matkaan eri kokoisilla syötteillä. Kuvasta näkee selkeästi kuinka Greedy-algoritmi on selkeästi näistä tarkin.