

Øving 8 Algoritmer og Datastrukturer

Lea Grønning og Ingebrigt Hovind

Programmet implementerer et javafx-kart, og pom.xml filen med nødvendige dependencies for dette er lagt ved. Visingen av kartet er nå kommentert ut, men dersom man ønsker å vise det etter å ha importert dependenciesene så kan man simpelthen av-kommentere linje 259 i Graph.java

Når man kjører programmet fra kommandolinja så må man føre inn følgende argumenter:

1: node fra, 2: node til, 3: node man vil finne stasjoner nær, 4: type stasjon og 5; antall stasjoner

Eksempel (trondheim til helsinki, ti ladestasjoner nær røros hotell)

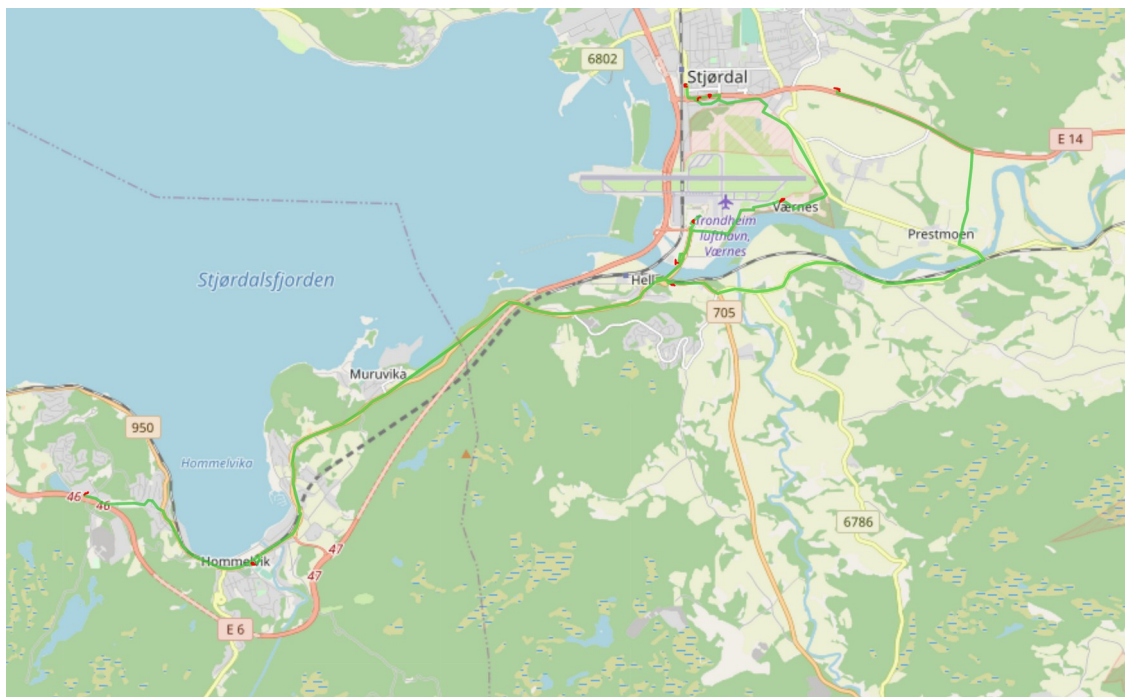
```
java Graph 2399829 1221382 1117256 4 10
```

Koden går ut ifra at javafilene ligger i samme mappe som kant-, node-, og interessepunktfilene

Bensinstasjon nær Trondheim lufthavn:

```
Dijkstra begynner søk etter nærmeste bensin/ladestasjoner  
Dijkstra fant 10 stasjoner, brukte 0 Millisekunder  
Behandlet 2499 noder
```

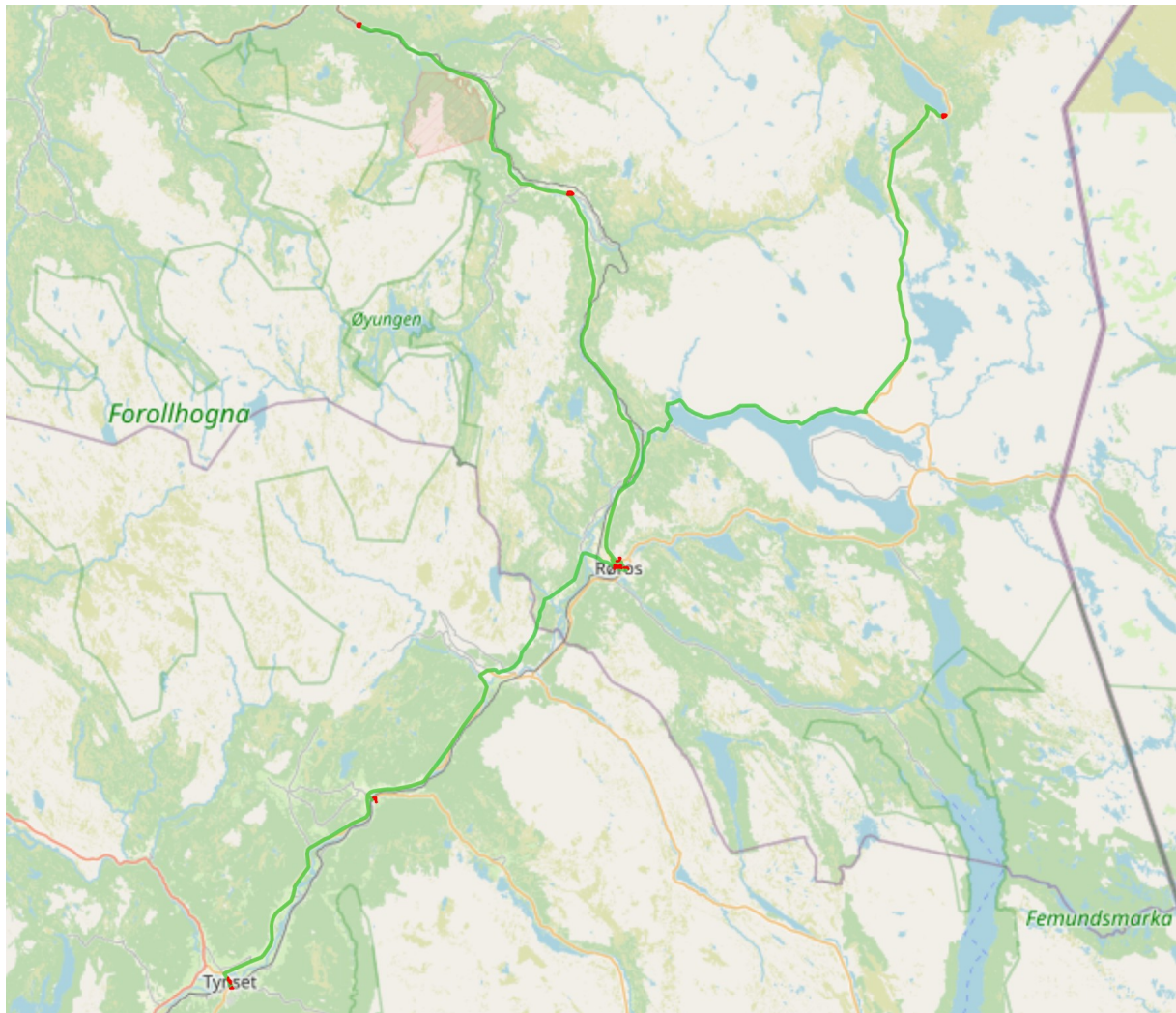
Noen av bensinstasjonene er vanskelig å se på kartet siden de ligger på veien til en av de andre bensinstasjonene. Bensinstasjonene er forstøkt markert rødt.



Ladestasjon nær «Røros hotell»:

```
Dijkstra begynner søk etter nærmeste bensin/ladestasjoner  
Dijkstra fant 10 stasjoner, brukte 0 Millisekunder  
Behandlet 15271 noder
```

Noen av ladestasjonene er vanskelig å se på kartet siden de ligger på veien til en av de andre ladestasjonene. Ladestasjonene er forsøkt markert rødt.



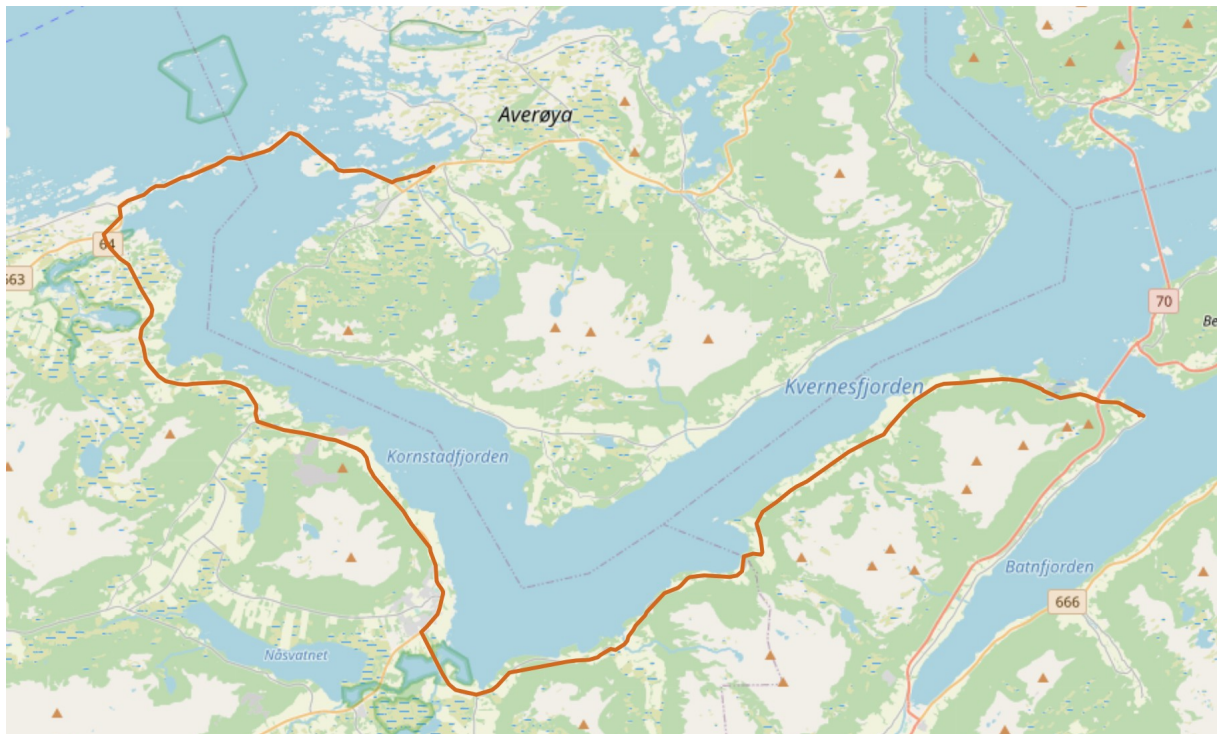
Dijkstras algoritme «Kårvåg-Gjemnes»:

Dijkstra leter etter korteste vei

Dijkstra fant korteste vei behandlet 11317 noder

Brukte: 32 millisekunder

Total reisetid med dijkstra er 0 Timer, 40 Minutter og 30 sekunder



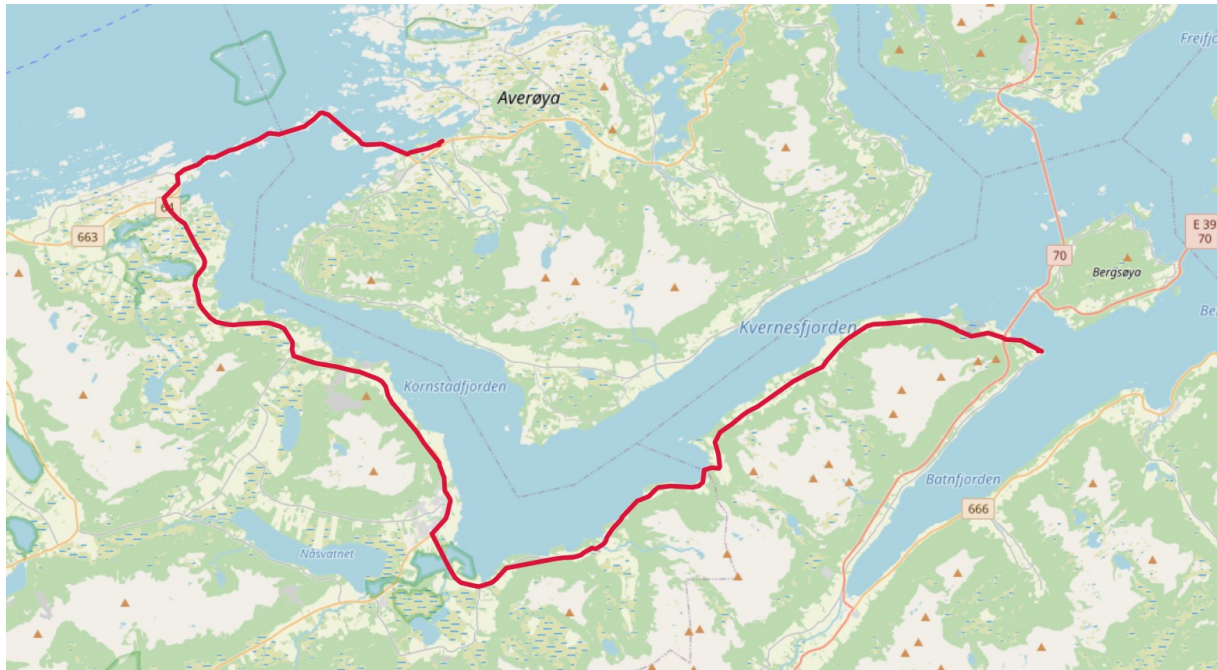
A* «Kårvåg-Gjemnes»:

```
A* leter etter korteste vei
```

```
A* fant korteste vei behandlet 7929 noder
```

```
Brukte: 15 millisekunder
```

```
Total reisetid med A* er 0 Timer, 40 Minutter og 30 sekunder
```



Om man sammenlikner Dijkstras algoritme og A* ser man at Dijkstra analyserte 11317 noder og brukte 32 millisekunder mens A* analyserte 7929 noder og brukte 15 millisekunder.

De begge fant samme vei med total reisetid på 40 minutter og 30 sekunder.

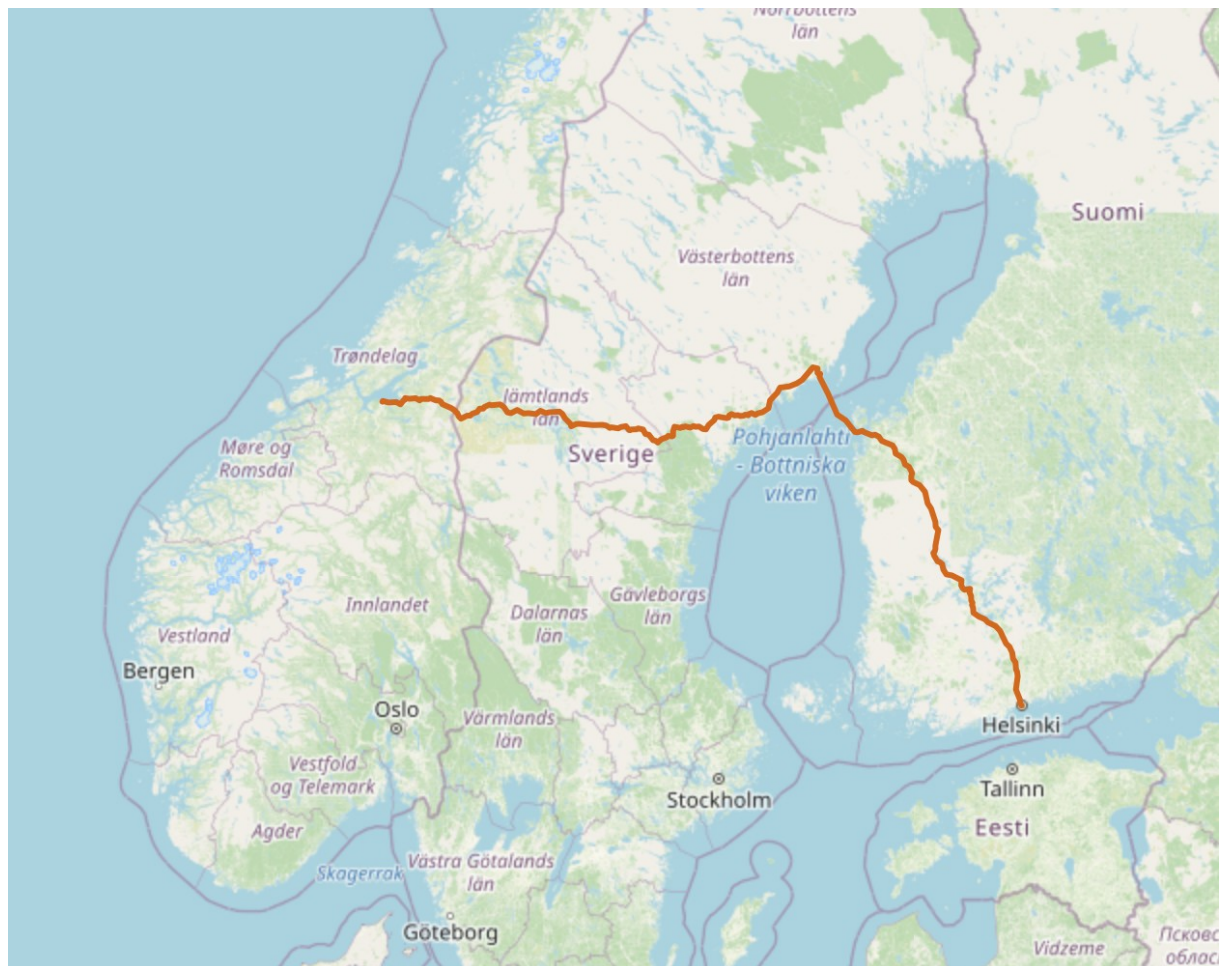
Dijkstras algoritme «Trondheim-Helsinki»:

Dijkstra leter etter korteste vei

Dijkstra fant korteste vei behandlet 5603735 noder

Brukte: 3968 millisekunder

Total reisetid med dijkstra er 15 Timer, 28 Minutter og 53 sekunder



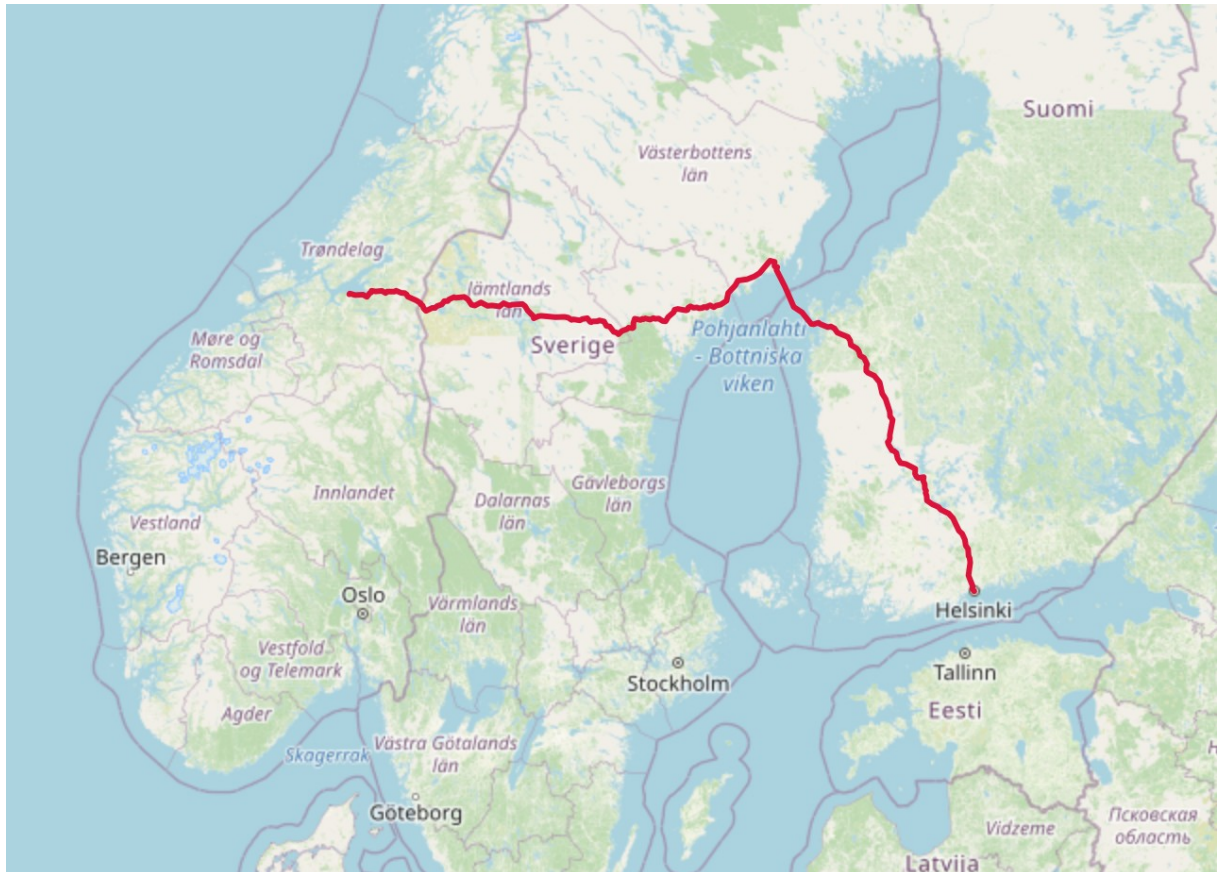
A* «Trondheim-Helsinki»:

A* leter etter korteste vei

A* fant korteste vei behandlet 2784842 noder

Brukte: 2257 millisekunder

Total reisetid med A* er 15 Timer, 28 Minutter og 53 sekunder



Om man sammenlikner Dijkstras algoritme og A* ser man at Dijkstra analyserte 5603735 noder og brukte 3968 millisekunder mens A* analyserte 2784842 noder og brukte 2257 millisekunder på Trondheim - Helsinki.

De begge fant samme vei med total reisetid på 15 timer 28 minutter og 53 sekunder.