

# A\* reitinhaku labyrintissa

## Testausdokumentti

### Koko ohjelman testaus:

Testataan sitä, kuinka paljon labyrintin koko ja sen monimutkaisuus vaikuttaa A\* -algoritmin nopeuteen. Vertailussa mukana astar-kansiossa olevat Labyrintti1, Labyrintti2, ..., Labyrintti9 tekstilabyrintit. Labyrintti3 ja Labyrintti5 ovat monimutkaisimmat ja suurimmat labyrintit. Testit suoritettu allekirjoittaneen kotoa löytyvällä tietokoneella ja vastaukset on tulostettu näytölle (eli Testi-luokka suoritettu yhdellä parametrilla).

Tuloksia:

<b>Labyrintti1: 0ms</b>	<b>Labyrintti6: ei ratkaisua</b>
<b>Labyrintti2: 0ms</b>	<b>Labyrintti7: 0ms</b>
<b>Labyrintti3: 0ms</b>	<b>Labyrintti8: 0ms</b>
<b>Labyrintti4: 0ms</b>	<b>Labyrintti9: ei ratkaisua</b>
<b>Labyrintti5: 0ms</b>	

Ohjelma löytää polun, mikäli sellainen on löydettävissä, testitapauksissa käytetyissä labyrinteissa nopeasti. Suoritusaikaa huomattavasti nostavien labyrinttien luominen käsin olisi jo liian suuri vaiva, joten tässä tapauksessa on tyydyttävä saatuun tulokseen jossa suoritusajakeroja ei havaittu.

### Keko-tietorakenteen testaus:

Testataan keon toimintaa antamalla sille eri määriä alkoita järjestettäväksi ja sitten tulostettavaksi suuruusjärjestyksessä pienin ensin. Testit suoritettu samalla tietokoneella kuin edellisekin. Keolle tehdyt testit voi toistaa Keko-luokasta löytyvän testausohjelman avulla.

<b>10-lukua</b>	<b>= 0ms</b>
<b>1000-lukua</b>	<b>= 30ms</b>
<b>10000-lukua</b>	<b>= 310ms</b>
<b>1000000-lukua</b>	<b>= 2550ms</b>
<b>10000000-lukua</b>	<b>= 281521ms</b>

Johtopäätöksenä keko toimii tyydyttävässä ajassa maltillisen kokoisella määrällä alkoita.