

## Mitä testattiin ja miten

Ohjelman testit ovat painottuneet lähinnä Junit-tyyppisiin testeihin, joissa tärkeää on ollut muun muassa tehokkuus, mutta myös algoritmien oikeellisuus. Käytännössä kaikki luokat ja metodit, jotka ovat tärkeitä algoritmin toimimiselle, on testattu. Testit ovat toteutettu palasissa, jolloin on saatu hyvät tulokset algoritmin oikeellisuudesta. Esimerkiksi A\*-algoritmin kaikki toiminnallisuus on testattu testaamalla siihen liittyviä metodeja, analysointia ja tehokkuutta. Myös omat tietorakenteet ovat testattuja myös ajallisesti.

Verkon luontia on myös testattu erikokoisilla syötteillä, joista suurin on ollut  $100 * 1000$ , koska tämän kokoisen verkon analysointi ja reitinhaku on sujunut alle yhdessä sekunnissa. Karttaa piirtäessä tosin on testattu lähinnä ajallisesti, että miten kauan polun generoimisessa menee ja tämä nopeus riippuu aivan siitä, että kuinka paljon hidastetta algoritmilta annetaan, jotta käyttäjällä olisi tarpeeksi aikaa klikata uusi maali algoritmilta. Hidasteen ollessa 0, toimii algoritmi ajassa  $O(|E| * |V| * \log|V|)$ .

## Minkälaisilla syötteillä testaus tehtiin

Testeissä on käytetty syötteenä erikokoisia karttoja, joissa on erilaisia esteitä hidasteena. Reitinhakuun liittyvissä testeissä näiden avulla on testattu polkujen oikeellisuutta, mutta myös polkujen generoinnin nopeutta ja tästä syystä jokaisessa testissä on ollut aikarajan 1 sekunti.

Testien karttoina on ollut mm. täysin tyhjiä kenttiä, joissa maali ja lähtö ovat vastakkaisissa kulmissa, tilanteita, jossa välissä on ollut [ muotoinen este sekä kartta, jossa on ollut L muotoinen este. Näistä kaikista haastavimmaksi on osoittautunut [ , joka on kohtisuorassa lähtöä kohti ja maalin välissä, koska tällöin A\* usein menee analysoimaan koko esteen välisen alueen.

Polkujen lyhyimmyyden määrittelemisessä on luotu ensin tyhjä kartta, jonka lyhyin reitti on laskettu käsin. Tämän jälkeen reittiin on lisätty esteitä ja laskettu lyhyin reitti uudelleen. Algoritmin on pitänyt löytää lyhyin reitti aina esteiden lukumäärästä riippumatta.

## Miten testit voidaan toistaa

Testit voidaan toistaa luomalla samanlaiset kartat, kuin testipakkauksessa tai luomalla omat kartat ja laskemalla ensin itse varmasti lyhyimmän reitin, jonka jälkeen algoritmi ajetaan ja tarkastetaan, että lyhyin polku on sama kuin se, mihin on itse tultu.

Muut testit voidaan toistaa käytännössä millä tahansa syötteillä, kunhan testattavat asiat ovat oikeita.

## Tulokset graafisena

\*kesken\*

## Suorituskykytestaus