

# MÄÄRITTELYDOKUMENTTI

## AIHE

Tarkoitus on toteuttaa ohjelma, joka analysoi eri reitinhakualgoritmien tehokkuuksia erilaisilla syötteillä. Ohjelmalle annetaan syötteenä (mustavalkoinen) kuvatiedosto sekä aloitus- ja lopetuspiste. Kuva määrittelee sokkelon, labyrintin tai muun alueen, jossa on esteitä (mustat pikselit) ja vapaata tilaa (valkoiset pikselit). Ohjelma ratkaisee lyhimmän reitin alku- ja loppupisteen välillä käyttäen useita eri algoritmeja ja mittaa niihin käytetyn suoritusajan.

Ohjelma palauttaa jokaiseen algoritmiin kuluneen ajan sekä luo uudet kuvatiedostot, joissa näkyy ratkaistut reitit sekä mahdollisesti muuta tietoa, kuten kaikki tarkistettut pikselit.

## ALGORITMIT JA TIETORAKENTEET

Ohjelmaan on tarkoitus toteuttaa useita ratkaisualgoritmeja, kuten esimerkiksi A\*, jump point search, dijkstra, breadth-first search, depth-first search (ei löydä aina lyhintä) sekä mahdollisesti muita, jos tulee lisää mieleen. Eri algoritmien sisällä voi vielä olla eri vaihtoehtoja etäisyysheuristiikalle, kuten esimerkiksi manhattan-etäisyys ja euklidinen etäisyys.

Tarvittaviin tietorakenteisiin kuuluu tietysti itse verkko, joka luetaan syötteestä ja sisältää ratkaistavan ongelman. Edellä mainittujen algoritmien toteuttamiseen tarvittavia tietorakenteita ovat ainakin min-heap, hash set, queue, stack sekä mahdollisesti erikoistuneempia rakenteita, kuten keko sisäisellä hajautustaululla sisältyvyyden tarkistuksen nopeuttamiseen.

## AIKA- JA TILAVAATIMUKSET

Algoritmista riippumatta suoritukseen kuluva aika riippuu täysin annetun syötteen laadusta (kuinka monimutkainen reitti? ratkaako ollenkaan?). Kaikissa algoritmeissa huonoimmassa tapauksessa (ei ratkaisua) aikavaativuus on  $O(|E|)$ , missä E on solmujen eli valkoisen pikseleiden määrä, sillä jokainen solmu on tutkittava.

## LÄHTEET

[https://en.wikipedia.org/wiki/List\\_of\\_algorithms#Graph\\_search](https://en.wikipedia.org/wiki/List_of_algorithms#Graph_search)