

Käyttöohje

Ohjelma löytyy repositorion juuresta .zip-tiedostona, nimellä **Pathfinding.zip** . Zipissä on ohjelma **pathfinding.jar** , ja se vaatii oheensa kansiot **mazes** ja **graphics** .

Oheistiedostot

mazes -kansio sisältää ohjelman käyttämät verkot eli sokkelot. Jokainen sokkelon on omassa kansiossaan.

Kansion tulee sisältää kolme tiedostoa:

- **maze.txt** - Sokkelo tekstitiedostona
- **startPosition.txt** - Aloitus sijainti koordinaatteina
- **targetMovement.txt** - Maalisijainti koordinaatteina sekä tämän jälkeen maalipisteen mahdollinen liike.

graphics -kansio sisältää visualisoinnin käyttämät grafiikat.

maze.txt

maze.txt kuvaa sokkelon merkkien avulla:

- - Normaali lattia
- **x** - Seinä (mikä tahansa tässä määrittelemätön merkki käy)
- **s** - Lattia, joka on suota. Liike suolla on kaksi kertaa raskaampaa.
- **i** - Lattia, joka on jäätä. Liike jäällä on puolet kevyempää.
- **d** - Ovi, joka on auki kun suoritus alkaa
- **D** - Ovi, joka on kiinni kun suoritus alkaa

On tärkeää, että sokkelotiedoston jokainen rivi on saman mittainen. Tällöin luotu verkko on nelikulmion mallinen ja käyttäytyy toivotusti.

startPosition.txt

Sisältää tiedon siitä, mistä ruudusta aloitetaan. Sisältää kaksi välilyönneillä erotettua kokonaislukua, jotka ovat aloitus sijainti x ja y -akseleilla. Vasen yläkulma on (0,0).

targetMovement.txt

Ensimmäinen rivi määrittää maalipisteen sijainnin samaan tapaan kuin `startPosition.txt` . Kaikki sen jälkeiset rivit tulkitaan maalipisteen liikkeiksi: rivillä on kaksi välilyönnillä erotettua kokonaislukua, joista ensimmäinen on liike x-akselilla ja toinen y-akselilla. Näin ollen rivi `-1 0` tarkoittaa yhden ruudun liikettä vasemmalle ja `1 1` liikettä yksi ruutu oikealle ja alas.

Ohjelman käynnistäminen

Ohjelma ajetaan komentoriviltä komennolla `java -jar pathfinding.jar` ja seuraavilla argumenteilla. Jos kaikkia argumentteja ei määritellä, ajetaan ohjelma oletusasetuksilla (määritetty tiedostossa `main.java`). Optimaalisessa tilanteessa, parhaiden tulosten saavuttamiseksi, ohjelma ajetaan ilman ylimääräisiä komentorivitulostuksia (`logEnabled = false`) ja ilman visualisointia (`windowEnabled = false`).

- `maze`
 - Käytettävä sokkelo. `mazes` -kansiossa olevan kansion nimi, esimerkiksi `41x41` .
- `allowDiagonal`
 - Onko vinottainen liike ruudukossa sallittu, `true` tai `false` .
- `algo`
 - Käytettävän algoritmin nimi. `A*` , `Dijkstra` , `Breadth-first` tai `Depth-first` .
- `moveDelay`
 - Monenko siirron välein maalipiste tekee oman siirtonsa, kokonaisluku.
 - Oletusarvo `5` .
- `doorDelay`
 - Monenko siirron välein ovet avautuvat tai sulkeutuvat, kokonaisluku.
 - Oletusarvo `20` .
- `waitBeforeFail`
 - Montako epäonnistunutta reitin löytämistä sallitaan peräkkäin ennen kuin operaatio todetaan toivottomaksi ja luovutetaan.
 - Tälle on tarvetta, jos sulkeutuvat ovet voivat aiheuttaa tilanteen jossa reittiä ei löydy tai jos maalipiste menee liikkueessaan seinän sisään. Tällöin odotetaan yhden siirron verran ja yritetään sitten uudestaan.
 - Oletusarvo `10` .

- `logEnabled`
 - Tulostetaanko jokaiseen polun uudelleenlaskentaan liittyvät tiedot komentoriville.
 - Etsimiseen käytetty aika ja askelten määrä.
 - `true` tai `false` , optimaalisten olosuhteiden tuottamiseksi `false` .
- `windowEnabled`
 - Käytetäänkö polunetsinnän visualisointia, `true` tai `false` .
 - Optimaalisten olosuhteiden tuottamiseksi `false` , sillä ikkuna hidastaa ohjelman toimintaa jonkin verran.
- `smallTiles`
 - Käytetäänkö visualisoinnissa 3x3 pikselin vai 10x10 pikselin ruutuja, `true` tai `false` .
 - Pienemmät ruudut ovat isompia sokkeloita varten.
- `waitInMillis`
 - Monenko millisekunnin välein uusi siirto tehdään, kokonaisluku.
 - Käytetään vain jos visualisointi on päällä. Ei vaikuta laskenta-ajan mittauksiin.
 - Oletusarvo `200` .

Lopullinen komento näyttää siis tältä: `java -jar pathfinding.jar t105x105 false A* 5 20 10 true true true 150`

Visualisoinnista

Käyttöliittymässä alkaa välittömästi pyöriä visualisaatio algoritmin toiminnasta. Eri väriset ruudut tarkoittavat seuraavaa:

- Punainen - takaa-ajaja
- Turkoosi - takaa-ajettu (maali)
- Musta - lattia
- Valkoinen - seinä
- Harmaa - ovi
- Kirkkaanvihreä - takaa-ajajan reitti
- Tummanvihreä - suo (isompi paino)
- Sininen - jää (matalampi paino)

Ohjelman tuottamat tulokset

Ohjelman käynnistyttyä se aloittaa valitun algoritmin suorittamisen valituilla parametreilla. Ohjelman suorituksen loputtua (visualisoinnin ollessa päällä sen pysähdyttyä) se kertoo tulokset komentorivillä (ja käyttöliittymän voi sulkea):

- Löytyikö reitti
- Laskemiseen käytetty aika yhteensä
- Käytettyjen askelten määrä (mukaanlukien odottamiset jos reittiä ei löydy)
- Yhteen askeleeseen käytetty keskimääräinen aika (ei välttämättä kovin hyödyllinen)
- Solmujen siivoamiseen käytetty aika (solmut siivotaan aina uudelleenlaskennan yhteydessä)
- Solmujen siivoamiseen käytetty keskimääräinen aika
- Laskentakertojen määrä (nimetty vähän hämäävästi recalculations)
- Keskimääräinen laskentaan käytetty aika
- Epäonnistuneet polunetsintäkerrat
- Pisin putki epäonnistuneita polunetsintöjä
- Kohdepisteen tekemät siirrot
- Ovien aukeamiset/sulkeutumiset
- Askeleet jäärävuodoissa (puolikas kaaren paino)
- Askeleet suoruvuodoissa (kaksinkertainen kaaren paino)

Kaikki ajat ovat millisekunneissa.

Jos komentoriviloki on päällä, tulostetaan joka uudelleenlaskentakerta:

- Laskentaan käytetty aika
- Tähänastinen laskentaan käytetty kokonaisaika
- Uuden polun pituus
- Laskemiseen käytetty aika per uuden polun askel (ei välttämättä hyödyllinen)