# PAC-MAN TOTEUTUSDOKUMENTTI

## Ohjelman yleisrakenne

A\* algoritmi koostuu Astar ja Anode luokista.

Algoritmia käyttää pääasiassa AbstractMoveLogic luokan toteuttavat luokat (BlinkyLogic, PinkyLogic, InkyLogic, ClydeLogic).

Algoritmi käyttää pacman.datastructure paketista löytyviä tietorakenteita.

## Saavutetut aika- ja tilavaativuudet

|  |  |
| --- | --- |
| Astar (G,w,a,b)  for kaikille solmuille v ∈ V  alkuun[v] = ∞  loppuun[v] = arvioi suora etäisyys v b  polku[v] = NIL  alkuun[a] = 0  S = ∅  while (solmu b ei ole vielä joukossa S)  valitse solmu u ∈ V \ S, jolle alkuun[v]+loppuun[v] on pienin  S = S ∪ {u}  for jokaiselle solmulle v ∈ vierus[u]  if alkuun[v] > alkuun[u] + w(u, v)  alkuun[v] = alkuun[u]+w(u, v)  polku[v] = u | |V| on kartan koko  O(|V|)  O(1)  O(1)  O(1)  O(1)  O(1)  O(|V|)  O(log |V|)  O(4), neljä ilmansuuntaa  O(1)  O(1)  O(1) |
| Tarkempi analyysi osoittaa että pahimman tapauksen aikavaativuus on O(|V| log 4).  Algoritmi pitää muistissa nykyistä karttaa. Kartta vie siis tilaa O(|V|).  Haettaessa lyhintä reittiä muodostetaan A\* solmutaulu jossa pidetään kirjaa etäisyydestä maaliin ja lähtöön, sekä edellisen solmun koordinaatit. Solmutaulun tilavaativuus on myöskin O(|V|)  Lisäksi muistiin tallennetaan vielä solmut joita on tarkasteltu, jotta pystytään toteuttamaan algoritmin visualisointi. Pahimmassa tapauksessa joudutaan tarkastelemaan koko kartta jolloin tilavaativuus on O(|V|). Siis pahimman tapauksen tilavaativuus on O(3 \* |V|) = O(|V|). | |

Minimikeko Lisäys ja poistamisoperaatioiden aikavaativuus O(log n) jossa n on keon koko. Koska Deepify ja Heapify komentojen suorittamiseen kuluu korkeintaan tämän verran aikaa. Muut operaation O(1). Tilavaativuus O(n) jossa n on pinon koko. Heapify operaation tilavaativuus on O(log n) rekursion takia. EnsureCapacity, jolla kasvatetaan keon kokoa vie hetkellisesti tilaa O(2\* keon koko). Ja aikavaativuus O(n) jossa n on keon alkioiden lukumäärä. Siis käytännössä aikavaativuus lisäysoperaatiolle on O(n) mutta EnsureCapacity suoritetaan pahimmassa tapauksessa O(log n) syötteeseen nähden jolloin operaatio tapahtuu harvakseltaan.

Pino Yksittäisten operaatioiden aikavaativuus O(1). Tilavaativuus O(n) jossa n on pinon koko. Yksittäisten operaatioiden tilavaativuus on O(1). Poikkeuksena EnsureCapacity, jolla kasvatetaan pinon kokoa vie hetkellisesti tilaa O(2\* pinon koko). Ja aikavaativuus O(n) jossa n on pinon alkioiden lukumäärä. Siis käytännössä aikavaativuus lisäysoperaatiolle on O(n) mutta EnsureCapacity suoritetaan pahimmassa tapauksessa O(log n) syötteeseen nähden jolloin operaatio tapahtuu harvakseltaan.

## Työn mahdolliset puutteet ja parannusehdotukset

Mahdollisuus käyttää samaa pistettä mikäli siihen tullaan eri suunnasta, ja siitä lähdetään eri suuntaan.

## Lähteet

Tietorakenteet ja algoritmit kurssin materiaali:  
<http://www.cs.helsinki.fi/u/floreen/tira2013syksy/tira.pdf>

The Pacman Dossier  
<http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>