

iMSc feladat 1

max 4 iMSc pont

Az A* algoritmus gyorsítására találták ki, hogy a problémater szélén található néhány csomópontra előzetesen kiszámítják az összes többi csomóponthoz vezető ténylegesen legrövidebb út távolságát.

Pl. egy $N=10000$ várossal rendelkező ország úthálózatánál körben az országhatáron kiválasztanak összesen $H=6$ várost, és mind az $N=10000$ város ezen $H=6$ városhoz vezető legrövidebb útját megkeresik. Ez persze nagy számítási teher, hiszen $6 \cdot 10000 = 6 \cdot 10^4$ távolságot kell meghatározni. (Viszont a kimerítő - minden csomópontpárt feldolgozó - számításhoz képest elenyésző, az $5 \cdot 10^7$ lenne.) Viszont ha ez az adat rendelkezésre áll, akkor a tényleges keresések majd gyorsíthatók, ezek az adatok heurisztikaként felhasználhatók.

Tehát egy lassú előzetes (off-line) vizsgálattal a tényleges (on-line) keresést kívánjuk gyorsítani.

Feladat

- Írjon egy olyan Java programot, és adja be futtatható formában, amely a következő bemeneteket kapja:
 - szomszedok.csv fájlban egy $N \cdot N$ mátrixot, amiben csak a közvetlenül szomszédos (pl. térképen úttal összekötött) csomópontok közt van megadva a távolság, a többi helyen 0-t tartalmaz. (Tehát, ha az 53-as csp szomszédos a 67-essel, akkor a $\text{szomszedok}(53,67) = \text{szomszedok}(67,53) > 0$, ha nem szomszédos, akkor $\text{szomszedok}(53,67) = \text{szomszedok}(67,53) = 0$),
 - hatarponttav.csv egy $H \cdot N$ -es mátrix, ahol az első sor az első határponttól vett tényleges optimális úttávolságokat tartalmazza, a második sor a második határcsomóponttól vett ténylegesen elérhető távolságokat stb.

A csv fájlok vesszővel (comma) elválasztva tartalmazzák a számértékeket. A program a futása elején bekér két sorszámot, megkeresi a két - megadott sorszámmal definiált - csomópont közt a legrövidebb utat, és megadja a kiértékelt csomópontok számával együtt.

- Hozzon létre egy, az eljárás tesztelésére alkalmas szoszedok.csv, hatarponttav.csv adathalmazt!
- Alkossa meg a heurisztikát, és a feladatban megadott két mátrix esetén (szomszedok.csv és hatrponttav.csv) vizsgálja meg, hogy az ezzel a heurisztikával végzett A*-mennyivel hatékonyabb az egyenletes költségű keresésnél!

Megjegyzések

- szomszedok.csp fájl megalkotását és a méretét a kijelölt konzulenssel beszélje meg, ehhez jelentkezzen Dobrowiecki Tadeusznál emailben (tade@mit.bme.hu)!
- a problémában említett algoritmus az A* keresés ún. ALT változata, amiről olvashat az alábbi elektronikus forrásokban:
 - Chapter 6, az alábbi riportban:
<https://www.microsoft.com/en-us/research/wp-content/uploads/2004/07/tr-2004-24.pdf>
 - Chapter 5, 6, az alábbi cikkben:
<http://www.siam.org/meetings/alnex05/papers/03agoldberg.pdf>
 - Chapter 2, az alábbi riportban:
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.396.6069&rep=rep1&type=pdf>
 - ALT algoritmus rész:
<https://logic.pdmi.ras.ru/midas/sites/default/files/midas-werneck.pdf>