

Úvod do Umelej Inteligencie Prehľadávanie grafu - UC & A*

October 2, 2025

PREHL'ADÁVANIE GRAFU

Budeme programovať prehľadacie algoritmy **Uniform-cost search** a **A***, pomocou ktorých budeme hľadať najkratšiu (v km) cestu medzi zastávkami Bratislavskej MHD.

Pozn.: V tejto úlohe neberieme do úvahy linky, ale iba spojenia medzi zastávkami (čiže nie "pôjdem 39nou, prestúpim na 4ku", ale "zo Zoo pôjdem na Lafranconi, odiaľ na Botanickú..."). Dáta taktiež neboli generované zo všetkých liniek, niektoré reálne spojenia v nich preto nie sú (napr. Zochova -> Hodžovo nám.).

Algorithm 1 - všeobecný algoritmus na prehľadávanie stromu (resp. grafu)

```
1: procedure UNIVERSAL SEARCH(start_node)
2:   Open ← start_node
3:   Explored ← {}
4:   while Open ≠ ∅ do
5:     N ← Open.pop()
6:     if goal_reached(N) then
7:       return N
8:     add N to Explored
9:     Add all neighbours of N to Open
10:  Search unsuccessful, finish
```

Rozdiel medzi DFS, BFS, Uniform-Cost, A* a ďalšími prehľadacími algoritmami je prakticky iba v tom, akú dátovú štruktúru použijeme pre OPEN list a ako do neho vkladáme vrcholy.

Program:

V súbore `ba_mhd_db.json` máte dáta - zastávky a ich spojenia. Samotný súbor nemusíte načítavať či parsovať, všetky metódy na to už máte hotové.

Kostru programu máte pripravenú v `bamhd.py`, vrátane príkladov ako používať priložené funkcie, odporúčame pozrieť si. Na reprezentáciu vrcholov **používajte pripravenú triedu `BusStop`**.

Naprogramujte Uniform-cost search a A* search na hľadanie najkratšej cesty (v km) medzi dvoma zastávkami. Ošetrte aj prípad, kedy cesta neexistuje, vtedy vráťte napr. `['Route not`

found']. Vypíšte tiež, koľko vrcholov grafu (zastávok) jednotlivé algoritmy pridali do *Open* zoznamu, a celkovú dĺžku cesty v km. (Tento výpis už je v kóde, iba ho nakrmte údajmi.)

Pre kontrolu: pre trasy "Zoo - Aupark"/"VW - Astronomicka" sú dĺžky ciest = 3.89km/18.08km.