

Tugas Kecil 3 IF2122 Strategi Algoritma

Implementasi Algoritma UCS dan A* untuk Menentukan Lintasan Terpendek

Tugas berkelompok, max 2 orang

Dikumpulkan: Rabu 12 April 2023 sebelum pukul 23.59

Tempat pengumpulan: Google Drive

Bahasa Pemrograman: C/C++/Java/Python/C#/Golang/Javascript/Ruby/Typescript/Kotlin

Platform: desktop/web based/mobile application

Algoritma UCS (*Uniform cost search*) dan A* (atau *A star*) dapat digunakan untuk menentukan lintasan terpendek dari suatu titik ke titik lain. Pada tugas kecil 3 ini, anda diminta menentukan lintasan terpendek berdasarkan peta Google Map jalan-jalan di kota Bandung. Dari ruas-ruas jalan di peta dibentuk graf. Simpul menyatakan persilangan jalan (simpang 3, 4 atau 5) atau ujung jalan. Asumsikan jalan dapat dilalui dari dua arah. Bobot graf menyatakan jarak (m atau km) antar simpul. Jarak antar dua simpul dapat dihitung dari koordinat kedua simpul menggunakan rumus jarak Euclidean (berdasarkan koordinat) atau dapat menggunakan *ruler* di Google Map, atau cara lainnya yang disediakan oleh Google Map.



Langkah pertama di dalam program ini adalah membuat graf yang merepresentasikan peta (di area tertentu, misalnya di sekitar Bandung Utara/Dago). Berdasarkan graf yang dibentuk, lalu program menerima input simpul asal dan simpul tujuan, lalu menentukan lintasan terpendek antara keduanya menggunakan algoritma UCS dan A*. Lintasan terpendek dapat ditampilkan pada peta/graf (misalnya jalan-jalan yang menyatakan lintasan terpendek diberi warna merah). Nilai heuristik yang dipakai adalah jarak garis lurus dari suatu titik ke tujuan.

Spesifikasi program:

1. Program menerima input *file* graf (direpresentasikan sebagai matriks ketetanggaan berbobot), jumlah simpul minimal 8 buah.
2. Program dapat menampilkan peta/graf
3. Program menerima input simpul asal dan simpul tujuan.
4. Program dapat menampilkan lintasan terpendek beserta jaraknya antara simpul asal dan simpul tujuan.
5. Antarmuka program bebas, apakah pakai GUI atau command line saja.

Bonus: Bonus nilai diberikan jika dapat menggunakan Google Map API untuk menampilkan peta, membentuk graf dari peta, dan menampilkan lintasan terpendek di peta (berupa jalan yang diberi warna). Simpul graf diperoleh dari peta (menggunakan API Google Map) dengan mengklik ujung jalan atau persimpangan jalan, lalu jarak antara kedua simpul dihitung langsung dengan rumus Euclidean.

Berkas yang dikumpulkan adalah laporan format PDF yang berisi:

1. Deskripsi persoalan
2. Kode program
3. Peta/graf input, output *screenshot* peta yang memperlihatkan lintasan terpendek untuk sepasang simpul, tampilkan hasil untuk beberapa lintasan terpendek di kota Bandung atau kota lainnya yang kalian suka.
4. Alamat github/google drive tempat kode sumber program diletakkan jika perlu dieksekusi oleh asisten.
5. Kesimpulan, komentar, dll

Peta jalan yang digunakan sebagai kasus uji adalah:

1. Peta jalan sekitar kampus ITB/Dago/Bandung Utara
2. Peta jalan sekitar Alun-alun Bandung
3. Peta jalan sekitar Buahbatu atau Bandung Selatan
4. Peta jalan sebuah kawasan di kota asalmu

Pranala pengumpulan laporan:

Kelas K1:

https://drive.google.com/drive/folders/1F-IPqotuVZlpzl_yONqrNf0VCTD6aw3S?usp=sharing

Kelas K2:

https://drive.google.com/drive/folders/1SDN178e_ofq8VCK0-rGXi9WXkkXivbvc?usp=sharing

Kelas K3:

<https://drive.google.com/drive/folders/1EcJcetWnBUy5HgfXOQ-IxFD3EMOiD8Vy?usp=sharing>

Lengkapi tabel berikut untuk memudahkan asisten menilai:

Centang (✓) jika ya

1	Program dapat menerima input graf	
2	Program dapat menghitung lintasan terpendek dengan UCS	
3	Program dapat menghitung lintasan terpendek dengan A*	
4	Program dapat menampilkan lintasan terpendek serta jaraknya	
5	Bonus: Program dapat menerima input peta dengan Google Map API dan menampilkan peta serta lintasan terpendek pada peta	

Pranala pengumpulan tugas

Lampiran: Menghitung jarak antar persimpangan dengan *ruler* di Google Map



