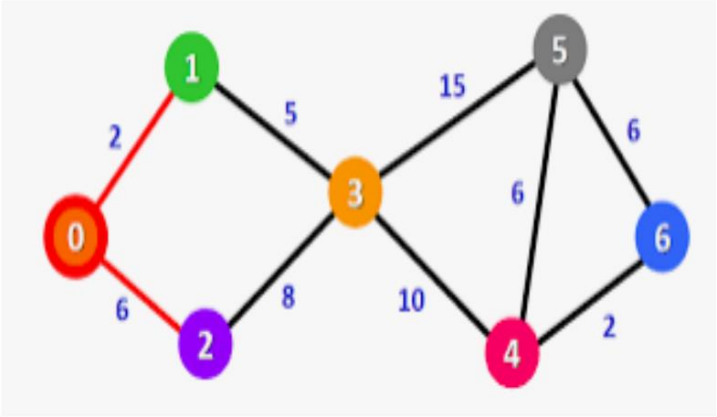
Nama : Muhammad Irfan Dharmalanga

NIM : G.231.22.0098

C. Dijkstra shortest path 2



Graph Yang Dipilih, Dijkstra algoritma

V	0	1	2	3	4	5	6
0	0	2	6	0	0	0	0
1	2	0	0	5	0	0	0
2	6	0	0	8	0	0	0
3	0	5	8	0	10	15	0
4	0	0	0	10	0	6	2
5	0	0	0	15	6	0	6
6	0	0	0	0	2	6	0

Nilai edges dari hubungan antar Simpul

- Cara kerja algoritma dijkstra

1. Inisialisasi graph menggunakan matriks ketetanggaan dengan bobot lintasan antar simpul.
2. Inisialisasi dist sebagai array yang menyimpan jarak terpendek dari source ke setiap simpul. Setel jarak source ke source menjadi 0, dan jarak ke simpul lain menjadi infinity (sys.maxsize).
3. Inisialisasi set sptSet sebagai set simpul yang telah diproses (shortest path tree). Semua elemen diatur False.
4. Lakukan loop sebanyak V (jumlah simpul) kali:
    a. Pilih simpul ’x’ yang belum diproses dengan jarak terpendek dari source.
    b. Tandai simpul ’x’ sebagai sudah diproses (sptSet[x] = True).
    c. Perbarui jarak terpendek ke semua simpul yang terhubung dengan simpul ’x’. Jika jarak baru lebih kecil dari jarak sebelumnya, perbarui nilai dist.
5. Cetak solusi berupa jarak terpendek dari source ke setiap simpul.

- Implementasi Dijkstra

6. Buat kelas Graph dengan metode __init__ untuk inisialisasi, printSolution untuk mencetak solusi, minDistance untuk menemukan simpul dengan jarak terpendek, dan dijkstra untuk menjalankan algoritma Dijkstra.
7. Buat objek graph g dengan 7 simpul.
8. Inisialisasi matriks ketetanggaan sebagai graf.
9. Panggil metode dijkstra pada objek g dengan simpul awal 0.
    - Catatan
10. Solusi berupa jarak terpendek dari source ke setiap simpul akan dicetak.
11. Graf dalam kode ini memiliki 7 simpul dan bobot lintasan antar simpulnya.
12. Implementasi menggunakan matriks ketetanggaan dan array untuk menyimpan jarak terpendek.

Hasil :

Simpul	Nilai
0	0
1	2
2	6
3	7
4	17
5	22
6	19