

TUGAS  
JARINGAN KOMPUTER



DISUSUN OLEH:

NAMA : Indah Frisilina Putri

NIM : 09011181419010

KELAS : SK 5A

FAKULTAS ILMU KOMPUTER  
JURUSAN SISTEM KOMPUTER  
UNIVERSITAS SRIWIJAYA

2016

## Algoritma Dijkstra

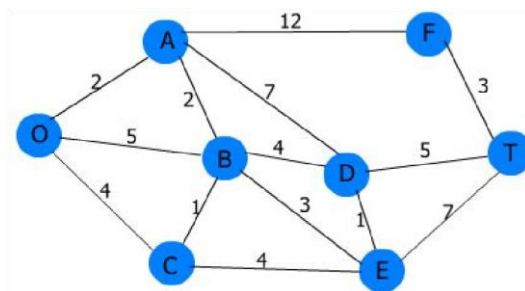
Algoritma Dijkstra adalah sebuah algoritma yang dikembangkan oleh seorang ilmuwan komputer dari Belanda, Edsger Dijkstra. Algoritma ini adalah sebuah algoritma yang menyelesaikan pencarian jalur terpendek pada graf dengan nilai non negatif untuk bobot setiap simpul, menghasilkan pohon jalur terpendek.

Penjelasan algoritma Dijkstra adalah :

1. Tetapkan nilai jarak pada setiap simpul. Tetapkan 0 untuk simpul awal dan tak terbatas pada semua simpul yang lain.
2. Tandai semua simpul sebagai belum dikunjungi. Tetapkan simpul sekarang sebagai simpul awal.
3. Untuk simpul sekarang, anggap semua tetangga yang belum dikunjungi dan hitung jarak terhadap simpul sekarang. Jika jarak sekarang lebih kecil dari jarak yang sebelumnya direkam, timpa nilainya.
4. Ketika kita selesai menghitung tetangga dari simpul sekarang, tandai sebagai telah dikunjungi. Jaraknya disimpan dan dinyatakan minimal.
5. Jika semua simpul telah dikunjungi, nyatakan sebagai selesai. Jika tidak, nyatakan simpul yang belum dikunjungi dengan jarak terkecil sebagai simpul sekarang dan ulangi langkah 3.

## Penyelesaian dengan Algoritma Dijkstra

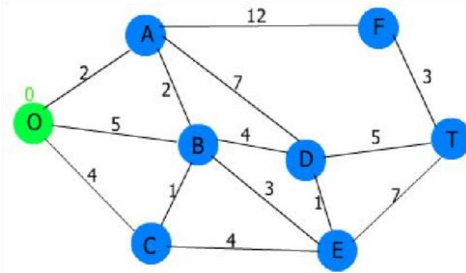
Terdapat sebuah kumpulan simpul dimana jarak diantara simpul dinyatakan pada nilai pada sisi yang berhubungan diantara mereka berdua.



Gambar 3. Sebuah graf yang memiliki keterkaitan antara yang satu dengan yang lain

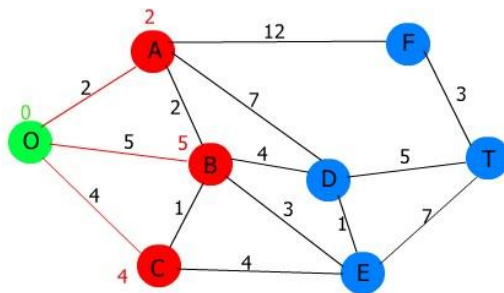
Permasalahannya adalah bagaimana mencari rute untuk menghasilkan jalur terpendek dari titik awal O ke titik akhir T ?

Pertama-tama, labelkan nilai simpul O dengan angka 0, seperti gambar berikut :



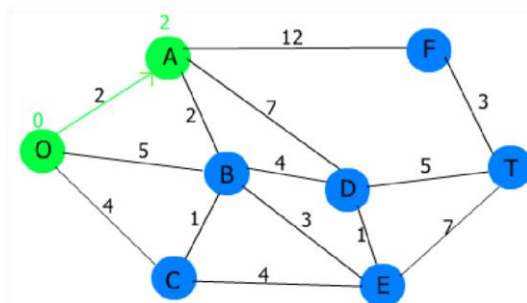
Gambar 4. Langkah pertama dalam algoritma Dijkstra

Lalu, identifikasi simpul-simpul mana yang belum dikunjungi, tapi terhubung dengan simpul awal, yaitu simpul O. Pada gambar, terlihat bahwa tetangga dari O adalah simpul A, B, dan C. Untuk setiap simpul yang memenuhi kriteria tersebut (simpul A,B, dan C), hitung jarak kandidat tersebut. Jarak kandidat = jarak menuju simpul + panjang sisi. Pilih simpul dengan bobot paling kecil.



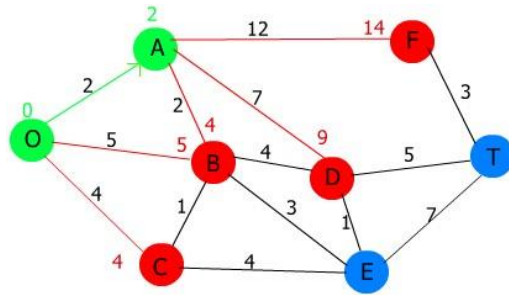
Gambar 5. Pemilihan simpul dengan bobot terkecil

Dari gambar 5, dapat diambil kesimpulan bahwa simpul A memiliki bobot minimum. Karena itu, tandai simpul A menjadi sudah dikunjungi dan labelkan dengan jarak kandidat. Tambahkan sudut ke kumpulan sudut.



Gambar 6. Pemilihan simpul terpendek

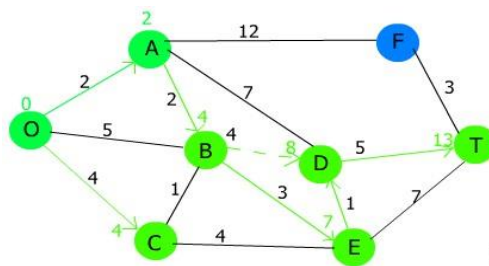
Identifikasi semua simpul yang belum teridentifikasi yang terhubung dengan sebuah simpul yang telah dikunjungi. Hitung semua jarak kandidat dari setiap sudut yang berhubungan.



**Gambar 7. Pemilihan nilai minimum dari simpul-simpul yang belum dikunjungi**

Terdapat 2 nilai yang sama, yaitu dari simpul O ke C dan simpul A ke B yang nilainya 4. Dalam kasus ini, kita pilih sembarang saja. Yang diambil sebagai simpul yang telah dikunjungi adalah B. Lakukan hal yang sama seperti sebelumnya, yaitu masukkan sudut AB ke dalam kumpulan sudut.

Lakukan sampai simpul T telah dikunjungi.



**Gambar 8. Simpul T telah masuk ke simpul yang telah dikunjungi**

Terdapat simpul yang belum dikunjungi, ada rute yang belum terselesaikan O-C. Terdapat 2 rute terpendek, yaitu : O-A-B-D-T, dan O-A-B-E-D-T. Keduanya memiliki bobot 13.