

TUGAS PEKAN 11 PENGANTAR REKAYASA DAN DESAIN

EDP CONCEPT SELECTION

1. Nomor Kelompok: 2
2. List Anggota:
 - a. Muhamad Salman Hakim Alfarisi (16521513)
 - b. Christophorus Dharma Winata (16521514)
 - c. Ahmad Rivai Yahya (16521523)
 - d. Wan Aupa Azis (16521525)
 - e. Jason Rivalino (16521541)
 - f. Afnan Edsa Ramadhan (16521542)
3. Judul projek: Pemetaan Kondisi Network Routing di ITB Jatinangor menggunakan Algoritma Dijkstra
4. Permasalahan yang akan dipecahkan:

Permasalahan yang akan dipecahkan dari tugas kelompok kami adalah mengenai kondisi pemetaan dari sistem *Router* yang ada di ITB Jatinangor. Kelompok kami akan melakukan observasi terkait dengan berbagai macam *router* yang tersebar di seluruh wilayah kompleks ITB Jatinangor. Dari berbagai router tersebut akan dicari kondisi yang memungkinkan terjadinya penerimaan sinyal yang paling optimal.

5. Input pada solusi:

Input yang ada dari tugas besar kelompok kami terdiri atas gambar peta dari ITB Jatinangor yang telah diberi keterangan posisi-posisi *Router*-nya, sehingga penggambaran jaringan internet yang ada di ITB Jatinangor menjadi jelas. Dari gambaran peta ini kemudian akan dicari jaringan yang paling optimal dalam kampus ITB Jatinangor dengan memanfaatkan algoritma Dijkstra dan program penentuan jaringan internet akan disusun dalam bahasa pemrograman Python.

Pada pengaplikasiannya, *router-router* akan dianggap sebagai node dan kemudian dihubungkan oleh *edge*. Pada setiap edge yang menghubungkan *node* (*router*) akan diberi nilai masing-masing sesuai dengan *cost* yang dimiliki. *Cost* akan merepresentasikan jarak antar *node* dan hambatan yang mungkin dialami oleh *node* dalam mengirimkan informasi (*packet*) kepada *node* yang lain. Semakin besar *cost* yang dimiliki oleh *edge*, maka akan semakin sulit pula dalam mengirimkan paket-paket informasi tersebut.

Selain itu, dalam kasus *routing* asli, terdapat pula input yang ada dalam setiap *router* yaitu alamat IP *router* yang bersifat unik yang berfungsi sebagai nama *node*-nya dan juga untuk pengguna (*user*) sendiri juga memiliki alamat IP yang bervariasi pada

setiap *device*-nya. Akan tetapi, dalam kajian tugas besar ini, router akan diwakilkan dengan nama yang sederhana untuk memudahkan pengerjaan tugas.

6. Output pada solusi:

Output yang akan dihasilkan pada akhir *routing* yaitu urutan node-node yang akan dilalui dengan cost paling rendah atau dengan yang aktif tercepat. Dalam kasus asli, juga dapat dihasilkan waktu yang ditempuh dari node asal ke node tujuan, namun dalam kajian ini akan dihasilkan cost total yang dibutuhkan untuk perjalanan dari node asal ke node tujuan. Berdasarkan hasil output inilah kemudian dapat ditemukan kondisi jaringan internet yang menghubungkan antara *router* dan *device* yang paling optimal.

7. Karakteristik dari solusi:

Karakteristik dari solusi sendiri berupa beberapa penjelasan yang dari solusi yang akan kelompok kami kaji untuk dipergunakan, diantaranya:

a. BFS (Breadth-First Search)

BFS merupakan metode pencarian alternatif solusi yang dilakukan dengan cara melakukan eksplorasi terlebih dahulu terhadap semua keadaan yang ada sebelum dilanjutkan pada level berikutnya. Pada metode ini, algoritma akan melakukan pengecekan terhadap peta secara melebar dan menyeluruh. Kemudian, sebuah simpul yang ada pada *graph* akan ditandai terlebih dahulu. Simpul yang berdekatan kemudian akan disimpan secara berurutan secara terus-menerus hingga mencapai simpul akhir sehingga *Graph* yang ada pada metode ini cenderung melebar dan lebih pendek. Berdasarkan hal ini, dapat disimpulkan bahwa BFS merupakan algoritma yang berbasis simpul (*vertex*).

b. DFS (Depth-First Search)

DFS merupakan metode pencarian alternatif solusi yang dilakukan dengan cara mengeksplorasi satu alternatif solusi hingga solusi yang ada berhasil didapatkan ataupun tidak berhasil sehingga harus mundur. Pada metode ini, algoritma lebih berbasis pada garis tepi (*edges*) dengan memanfaatkan *stack* untuk melakukan penyimpanan pada garis simpul yang dikunjungi. Metode ini bekerja secara rekursif dengan eksplorasi simpul yang dilakukan pada sepanjang garis tepi. Pencarian dilakukan dengan melakukan eksplorasi terhadap simpul-simpul yang belum ditemukan sehingga *Graph* yang ada pada metode ini cenderung menyempit dan memanjang.

c. Algoritma Uniform Cost Search (UCS) atau A* (A-Star)

A* (A-Star) merupakan algoritma yang digunakan untuk mendapat solusi yang dikembangkan dengan memanfaatkan fungsi heuristik didalamnya sehingga pencarian algoritma dapat dilakukan dengan lebih cepat dan efisien akan tetapi tidak terlalu lengkap dan optimal. Pada *graph* ini akan dikombinasikan antara nilai jalur

dan angka dan kemudian akan dicari nilai akhir terendah yang mungkin didapatkan dari semua simpul yang telah dilalui dan semua jalur yang telah dieksplorasi berdasarkan perhitungan. Nilai akhir yang akan dicari dapat dinotasikan sebagai berikut:

$$F(n) = g(n) + h(n)$$

Keterangan:

$F(n)$: nilai akhir

$g(n)$: nilai perjalanan dari simpul awal ke simpul n

$h(n)$: nilai heuristic dari simpul n menuju simpul akhir

d. Algoritma Bellman Ford

Algoritma Bellman Ford adalah algoritma pencarian shortest path untuk graf yang terdapat bobot edge yang bernilai negatif. Algoritma ini mempunyai kemiripan dengan algoritma Dijkstra. Terdapat simpul/node yang dihubungkan dengan edge berarah. Hal yang membedakan adalah algoritma Bellman Ford dapat bekerja pada edge yang bernilai negatif, sedangkan algoritma Dijkstra tidak.

e. Algoritma Dijkstra

Dijkstra merupakan algoritma yang dipakai untuk menyelesaikan permasalahan jarak terpendek dari sebuah *graph* yang berarah. Cara kerja dari algoritma ini adalah dengan membuat jalur ke satu simpul yang optimal pada setiap langkahnya. Sehingga, pada langkah tertentu sudah diketahui beberapa jalur terpendeknya.

8. Tabel Concept Selection solusi (Concept Scoring):

		Concepts										
		BFS			DFS		A* (A-Star)		Bellman - Ford		Dijkstra	
Selection Criteria	Weight	Rating	Score	Rating	Score	Rating	Score	Rating	Score	Rating	Score	
Keefektifan dalam mencari koneksi internet tercepat	45%	3	1,35	1	0,45	3	1,35	4	1,8	5	2,25	
Banyak memori yang dipergunakan	25%	2	0,5	4	1	3	0,75	3	0,75	3	0,75	
Kemudahan dalam penggunaan	15%	3	0,45	3	0,45	2	0,3	2	0,3	2	0,3	
Lama waktu pencarian routing (Running Time)	10%	3	0,3	4	0,4	3	0,3	2	0,2	2	0,2	
Kemudahan dalam penyusunan algoritma	5%	3	0,15	3	0,15	2	0,1	2	0,1	2	0,1	
Total score		2,75		2,45		2,8		3,15		3,6		
Rank		4		5		3		2		1		
Continue		No		No		No		No		Yes		