

Branch and Bound untuk Rute Terpendek Tur Pengenalan Labtek V Gedung Benny Subianto

Chita Najmi Nabila - 13509015
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganessa 10 Bandung 40132, Indonesia
13509015@std.stei.itb.ac.id

Abstrak – Di Institut Teknologi Bandung, mahasiswa program studi akan dikelompokkan ke dalam himpunan kemahasiswaan. Himpunan mahasiswa ini didirikan agar menjadi wadah pembelajaran bagi mahasiswa di program studi yang bersangkutan. Untuk bergabung ke dalam himpunan, mahasiswa harus melalui tahapan kaderisasi yang telah ditetapkan oleh masing-masing himpunan. Mahasiswa informatika di ITB menetapkan Simulasi dan Pelatihan Keorganisasian untuk Anggota (SPARTA) sebagai jalan masuk mahasiswa baru ke dalam himpunan. Dalam SPARTA, calon anggota himpunan akan diperkenalkan kepada lingkungan keprofesian informatika mencakup tur mengelilingi Labtek 5 Gedung Benny Subianto sebagai ‘sarang’ mahasiswa informatika. Untuk memudahkan pencarian rute terpendek pelaksana SPARTA dalam memperkenalkan Labtek V kepada mahasiswa baru, akan digunakan algoritma *Branch and Bound* dengan menggunakan skema *Breadth First Search* pada permasalahan *Traveling Salesman Problem*. Untuk memudahkan penulisan, maka yang digunakan pada contoh kasus di makalah ini hanya tidak menggunakan seluruh *spot* di Labtek V.

Kata kunci – rute terpendek, *Branch and Bound*, tur mengelilingi Labtek V, *Traveling Salesman Problem*, *Breadth First Search*

I. PENDAHULUAN

Setiap himpunan mahasiswa di Institut Teknologi Bandung memiliki cara yang berbeda-beda dalam menurunkan nilai-nilai kaderisasi kepada mahasiswa baru. Himpunan Mahasiswa Informatika juga memiliki cara tersendiri dalam mewadahi mahasiswa baru dalam menjajaki dunia keprofesian informatika.

Salah satu metode yang digunakan oleh Himpunan Mahasiswa Informatika adalah dengan melakukan kegiatan keorganisasian dan kemahasiswaan berbasis simulasi. Pada kegiatan simulasi ini, mahasiswa baru akan diperkenalkan dengan seluk beluk informatika, termasuk di dalamnya tempat mahasiswa informatika menuntut ilmu sekaligus berorganisasi, Labtek V Gedung Benny Subianto.

Pada makalah ini, penulis akan melakukan pembahasan terhadap pencarian rute pengenalan Labtek V kepada mahasiswa baru agar tidak memakan waktu yang lama. Permasalahan ini tergolong ke dalam *Traveling Salesman Problem* dan akan ditangani dengan algoritma *Branch*

and Bound.

II. DASAR TEORI

Algoritma *Branch and Bound* – berikutnya disebut B&B – adalah salah satu metode pencarian di dalam ruang solusi secara sistematis. Ruang solusi diorganisasikan ke dalam pohon ruang status yang dibangun dengan skema *Breadth First Search* – berikutnya disebut BFS. Untuk mempercepat pencarian solusi pada simpul tertentu, maka setiap simpul diberikan sebuah nilai *cost* di mana simpul berikutnya akan diekspansi berdasarkan *cost* tersebut.

Pada algoritma BFS, semua simpul yang dibangkitkan akan diantrikan di belakang antrian. Konsep dasar dari algoritma BFS ini adalah *First In First Out* (FIFO), di mana ketika simpul aktif sudah masuk ke dalam antrian, simpul berikutnya yang dapat masuk adalah simpul pertama yang mengantri.

Dengan menggunakan algoritma BFS, simpul-simpul yang sudah dibangkitkan tidak akan diekspansi lagi sehingga tidak terjadi pengulangan pembangkitan simpul. Nilai *cost* pada setiap simpul menyatakan taksiran *cost* minimum dari simpul awal ke simpul solusi. Nilai tersebut juga menyatakan batas bawah (*lower bound*) dari *cost* pencarian simpul solusi. *Cost* ini sendiri akan dikalkulasi dengan menggunakan fungsi pembatas yang mampu membatasi simpul yang tidak mengarah ke simpul solusi.

II. 1 Algoritma *Branch and Bound*

Pada algoritma B&B, pencarian simpul solusi dapat dipercepat dengan memilih simpul aktif berdasarkan *cost*-nya. Setiap simpul aktif diasosiasikan dengan sebuah ongkos yang menyatakan nilai batas (*bound*) yang dapat berupa:

- jumlah simpul pada *sub*-pohon yang perlu dibangkitkan sebelum simpul solusi ditemukan, dan
- panjang lintasan lintasan dari suatu simpul ke simpul solusi terdekat.

Berikut ini adalah urutan kerja algoritma B&B secara umum:

- Masukkan simpul akar ke dalam antrian.
- Jika simpul akar adalah simpul solusi, maka solusi ditemukan dan algoritma berhenti bekerja.

3. Jika antrian kosong, maka solusi tidak ada dan algoritma berhenti bekerja.
4. Jika antrian tidak kosong, pilih sebuah simpul sembarang pada antrian yang memiliki *cost* minimum.
5. Jika simpul tersebut adalah simpul solusi, maka solusi ditemukan dan algoritma berhenti bekerja.
6. Jika simpul tersebut bukan simpul solusi, maka periksa apakah simpul tersebut memiliki anak.
7. Jika simpul tidak memiliki anak, kembali ke proses 3 (tiga).
8. Jika simpul tersebut memiliki anak hitung *cost* dari simpul anak dan masukkan simpul anak ke dalam antrian.
9. Proses selanjutnya berulang dimulai dari proses 3 (tiga).

II. 2 Traveling Salesman Problem

Persoalan TSP adalah salah satu permasalahan di dunia nyata yang masih sulit ditemukan komputasi penyelesaiannya yang efektif dan efisien. Pada dasarnya, masalah utama TSP adalah bagaimana seorang *salesman* dapat menghampiri seluruh kota untuk menjajakan barangnya melalui rute terpendek.

Dengan algoritma *Brute Force*, rute terpendek dapat dihitung dengan cara melakukan perhitungan seluruh kemungkinan perjalanan *salesman*. Namun dengan begitu akan terjadi masalah ketika kita menemukan persoalan dengan kota dalam jumlah yang sangat besar. Hal ini tentu saja mengurangi efektivitas waktu.

III. ALGORITMA BRANCH AND BOUND SEBAGAI SOLUSI TRAVELING SALESMAN PROBLEM

Dalam sebuah persoalan TSP, misalkan diberikan sebuah graf $G = (V, E)$ yang merepresentasikan sebuah graf rute tur yang mungkin ditempuh.

Keterangan:

- $|V|$ adalah jumlah simpul dalam graf G di mana masing-masing simpul akan diberi nama A, B, C, ..., n.
- c_{ij} adalah jarak rute yang harus ditempuh dari suatu simpul i ke simpul lainnya.
- Tur dilakukan berawal dari simpul A.
- S adalah ruang solusi dengan $S = \{(1, \pi, 1) \mid \pi \text{ adalah permutasi simpul}\}$.
- $|S|$ adalah banyaknya solusi yang mungkin diperoleh dengan nilai $|S| = (n - 1)!$

Solusi TSP sendiri dinyatakan dengan bentuk vektor $X = (x_0, x_1, x_2, \dots, x_{n-1}, x_n)$ dengan $x_0 = x_n =$ simpul awal dan akhir dari Tur Pengenalan Labtek V Gedung Benny Subianto, dalam hal ini adalah simpul A, Sekretariat 1 HMIF.

Cost untuk setiap simpul dapat dihitung dengan melakukan pemetaan simpul ke dalam matriks dan menggunakan matriks tereduksi (*reduced cost matrix*) dari graf G . Matriks tereduksi adalah matriks yang memiliki paling tidak satu buah nilai 0 (nol) di setiap

baris dan kolomnya. Selain itu, tidak ada nilai negatif pada matriks ini.

Sebuah matriks hasil reduksi akan menjadi acuan pada perhitungan berikutnya. Dengan begitu matriks akan mengalami reduksi pada saat pencarian rute selama kondisi tidak terpenuhi. Berikut ini adalah cara melakukan reduksi *cost* matriks pada saat pencarian rute terpendek tur:

1. Mengubah nilai pada baris i dan kolom j menjadi ∞ .
2. Mengubah nilai elemen pada kolom pertama di baris j menjadi ∞ .
3. Melakukan reduksi kembali matriks yang menjadi acuan selanjutnya.

Apabila dilakukan perumusan, maka akan didapatkan nilai dari fungsi pembatas elemen pada matriks yaitu:

$$\hat{c}(S) = \hat{c}(R) + A(i, j) + r$$

Keterangan:

- $\hat{c}(S)$: *Cost* minimum melalui simpul S
- $\hat{c}(R)$: *Cost* minimum melalui simpul R (yang merupakan *parent* dari simpul S)
- $A(i, j)$: *Cost* antar simpul i dengan simpul j
- R : Jumlah nilai reduksi pada matriks di simpul S

IV. PENCARIAN RUTE TERPENDEK UNTUK TUR PENGENALAN GEDUNG BENNY SUBIANTO

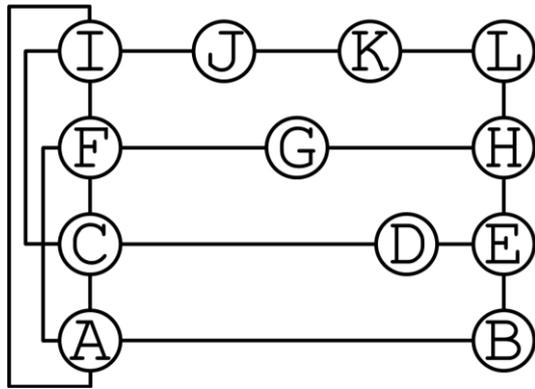
Sebagaimana yang telah disebutkan pada bab pendahuluan, kita akan mencari rute terpendek pada saat tur pengenalan Labtek V Gedung Benny Subianto kepada mahasiswa baru.



Gambar 1 Labtek V Gedung Benny Subianto

IV.1 Penentuan Spot Tur sebagai Objek TSP

Spot yang akan digunakan dalam uji coba pada makalah ini akan tergambar sebagai berikut:



Gambar 2 Spot-Spot yang Dikunjungi

Keterangan graf:

- Simpul A : Sekretariat 1I HMIF
- Simpul B : Selasar Labtek V
- Simpul C : Area Laboratorium 1
- Simpul D : Ruang Ketua Program Studi
- Simpul E : Selasar TU IF
- Simpul F : Ruang Kuliah
- Simpul G : Area Laboratorium 2
- Simpul H : Selasar MIC
- Simpul I : Area Laboratorium 3
- Simpul J : Dapur Labtek V

- Simpul K : Area Laboratorium Dasar
- Simpul L : Sekretariat 2 HMIF

Dari graf tersebut, kita juga bisa membuat matriks *cost* antar *spot* yang merepresentasikan jarak antar *spot* tersebut, yaitu:

∞	12	5	X	X	12	X	X	20	X	X	X
12	∞	X	X	5	X	X	X	X	X	X	X
5	X	∞	9	X	5	X	X	12	X	X	X
X	X	9	∞	3	X	X	X	X	X	X	X
X	5	X	3	∞	X	X	5	X	X	X	X
12	X	5	X	X	∞	6	X	5	X	X	X
X	X	X	X	X	6	∞	6	X	X	X	X
X	X	X	X	5	X	6	∞	X	X	X	5
20	X	12	X	X	5	X	X	∞	4	X	X
X	X	X	X	X	X	X	X	4	∞	4	X
X	X	X	X	X	X	X	X	X	4	∞	4
X	X	X	X	X	X	X	X	5	X	X	4
X	X	X	X	X	X	X	X	5	X	X	4
X	X	X	X	X	X	X	X	5	X	X	4
X	X	X	X	X	X	X	X	5	X	X	4
X	X	X	X	X	X	X	X	5	X	X	4

IV.2 Pencarian Cost Rute Terdekat

Reduksi matriks yang dilakukan adalah:

	A	B	C	D	E	F	G	H	I	J	K	L
A	∞	12	5	X	X	12	X	X	20	X	X	X
B	12	∞	X	X	5	X	X	X	X	X	X	X
C	5	X	∞	9	X	5	X	X	12	X	X	X
D	X	X	9	∞	3	X	X	X	X	X	X	X
E	X	5	X	3	∞	X	X	5	X	X	X	X
F	12	X	5	X	X	∞	6	X	5	X	X	X
G	X	X	X	X	X	6	∞	6	X	X	X	X
H	X	X	X	X	5	X	6	∞	X	X	X	5
I	20	X	12	X	X	5	X	X	∞	4	X	X
J	X	X	X	X	X	X	X	X	4	∞	4	X
K	X	X	X	X	X	X	X	X	X	4	∞	4
L	X	X	X	X	X	X	X	5	X	X	4	∞

RA - 5
RB - 5
RC - 5
RD - 3
RE - 3
RF - 5
RG - 6
RH - 5
RI - 4
RJ - 4
RK - 4
RL - 4

	A	B	C	D	E	F	G	H	I	J	K	L
A	∞	7	0	X	X	7	X	X	15	X	X	X
B	7	∞	X	X	0	X	X	X	X	X	X	X
C	0	X	∞	4	X	0	X	X	7	X	X	X
D	X	X	6	∞	0	X	X	X	X	X	X	X
E	X	2	X	0	∞	X	X	2	X	X	X	X
F	7	X	0	X	X	∞	1	X	0	X	X	X
G	X	X	X	X	X	0	∞	0	X	X	X	X
H	X	X	X	X	0	X	1	∞	X	X	X	0
I	16	X	8	X	X	1	X	X	∞	0	X	X
J	X	X	X	X	X	X	X	X	0	∞	0	X
K	X	X	X	X	X	X	X	X	X	0	∞	0
L	X	X	X	X	X	X	X	1	X	X	0	∞

CA - 0
CB - 2
CC - 0
CD - 0
CE - 0
CF - 0
CG - 1
CH - 0
CI - 0
CJ - 0
CK - 0
CL - 0

	A	B	C	D	E	F	G	H	I	J	K	L
A	∞	5	0	X	X	7	X	X	15	X	X	X
B	7	∞	X	X	0	X	X	X	X	X	X	X
C	0	X	∞	4	X	0	X	X	7	X	X	X
D	X	X	6	∞	0	X	X	X	X	X	X	X
E	X	0	X	0	∞	X	X	2	X	X	X	X
F	7	X	0	X	X	∞	0	X	0	X	X	X
G	X	X	X	X	X	0	∞	0	X	X	X	X
H	X	X	X	X	0	X	0	∞	X	X	X	0
I	16	X	8	X	X	1	X	X	∞	0	X	X
J	X	X	X	X	X	X	X	X	0	∞	0	X
K	X	X	X	X	X	X	X	X	X	0	∞	0
L	X	X	X	X	X	X	X	1	X	X	0	∞

Matriks terakhir adalah matriks yang akan dijadikan acuan dalam ekspansi dan reduksi isi matriks sehingga kita bisa mendapatkan *cost*-nya.

$$\begin{aligned} \hat{c}(\text{root}) &= (5+5+5+3+3+5+6+5+4+4+4+4) + (2+1) \\ &= 53 + 3 \\ &= 56 \end{aligned}$$

Selanjutnya, kita akan melakukan ekspansi simpul akar

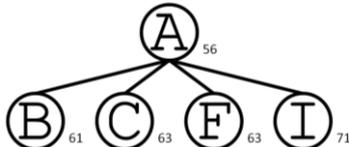
dan mendapatkan simpul anak-anaknya dengan penomoran sesuai dengan penentuan *spot* pada pembahasan sebelumnya.

Perhitungan rute simpul-simpul sesuai dengan perumusan yang sudah dijelaskan, diawali dengan ekspansi di simpul A:

1. Simpul B; Lintasan A, B
 $r = 0; \hat{c}(B) = 56 + 5 + 0 = 61$

- Simpul C; Lintasan A, C
 $r = 7; \hat{c}(C) = 56 + 0 + 7 = 63$
- Simpul F; Lintasan A, F
 $r = 0; \hat{c}(F) = 56 + 7 + 0 = 63$
- Simpul I; Lintasan A, I
 $r = 0; \hat{c}(I) = 56 + 15 + 0 = 71$

Cost minimum yang kita dapat adalah 61 (enam puluh satu) sehingga simpul selanjutnya yang akan diekspansi adalah simpul B.

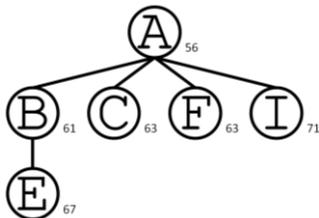


Gambar 2 Pohon Ruang Status Hasil Ekspansi Simpul Root A

Selanjutnya, dilakukan ekspansi pada simpul B:

- Simpul E; Lintasan A, B, E
 $r = 6; \hat{c}(E) = 61 + 0 + 6 = 67$

Cost minimum yang kita dapat adalah 67 (enam puluh tujuh) sehingga simpul selanjutnya yang akan diekspansi adalah simpul E.

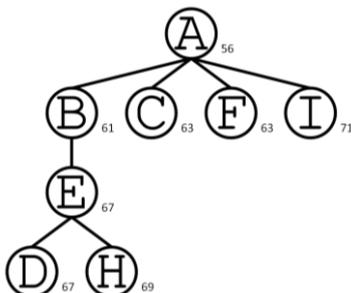


Gambar 3 Pohon Ruang Status Hasil Ekspansi Simpul B

Selanjutnya, dilakukan ekspansi pada simpul E:

- Simpul D; Lintasan A, B, E, D
 $r = 0; \hat{c}(D) = 67 + 0 + 0 = 67$
- Simpul H; Lintasan A, B, E, H
 $r = 0; \hat{c}(H) = 67 + 2 + 0 = 69$

Cost minimum yang kita dapat adalah 67 (enam puluh tujuh) sehingga simpul selanjutnya yang akan diekspansi adalah simpul D.

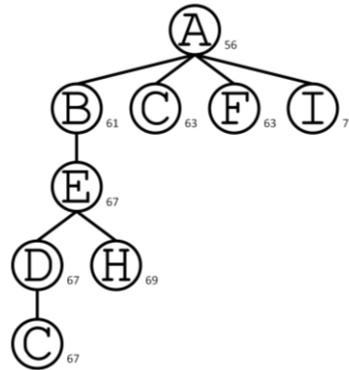


Gambar 4 Pohon Ruang Status Hasil Ekspansi Simpul E

Selanjutnya, dilakukan ekspansi pada simpul D:

- Simpul C; Lintasan A, B, E, D, C
 $r = 0; \hat{c}(L) = 67 + 0 + 0 = 67$

Cost minimum yang kita dapat adalah 67 (enam puluh tujuh) sehingga simpul selanjutnya yang akan diekspansi adalah simpul C.

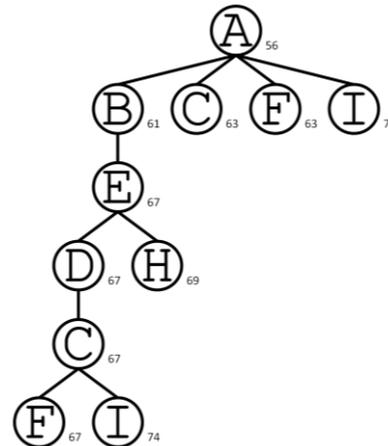


Gambar 5 Pohon Ruang Status Hasil Ekspansi Simpul D

Selanjutnya, dilakukan ekspansi pada simpul C:

- Simpul F; Lintasan A, B, E, D, C, F
 $r = 0; \hat{c}(F) = 67 + 0 + 0 = 67$
- Simpul I; Lintasan A, B, E, D, C, I
 $r = 0; \hat{c}(I) = 67 + 7 + 0 = 74$

Cost minimum yang kita dapat adalah 67 (enam puluh tujuh) sehingga simpul selanjutnya yang akan diekspansi adalah simpul F.

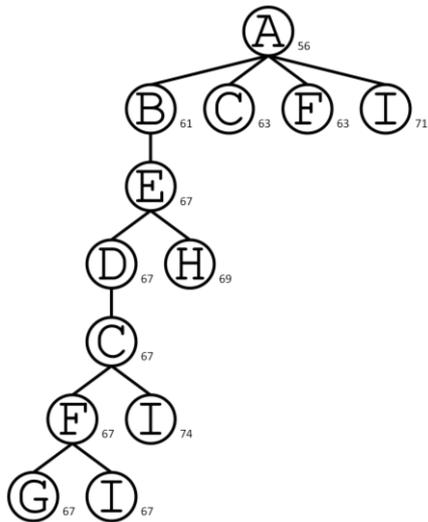


Gambar 6 Pohon Ruang Status Hasil Ekspansi Simpul C

Selanjutnya, dilakukan ekspansi pada simpul F:

- Simpul G; Lintasan A, B, E, D, C, F, G
 $r = 0; \hat{c}(G) = 67 + 0 + 0 = 67$
- Simpul I; Lintasan A, B, E, D, C, F, I
 $r = 0; \hat{c}(I) = 67 + 0 + 0 = 67$

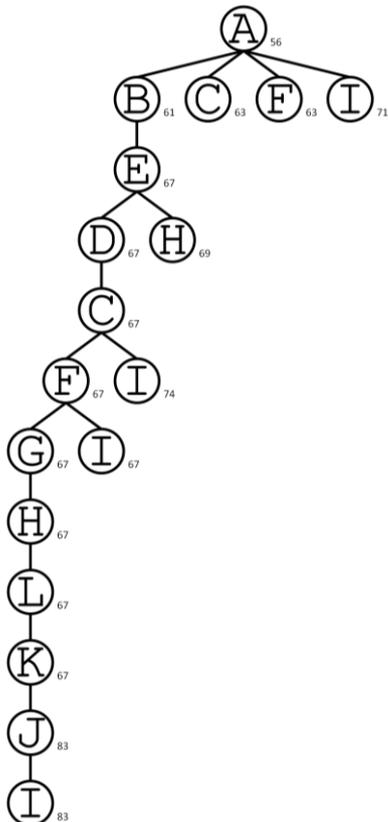
Karena hasil perhitungan cost sama yaitu 67 (enam puluh tujuh), maka kira akan melakukan ekspansi terhadap kedua simpul G dan simpul I.



Gambar 7 Pohon Ruang Status Hasil Ekspansi Simpul F

Selanjutnya, dilakukan ekspansi pada simpul G sampai selesai:

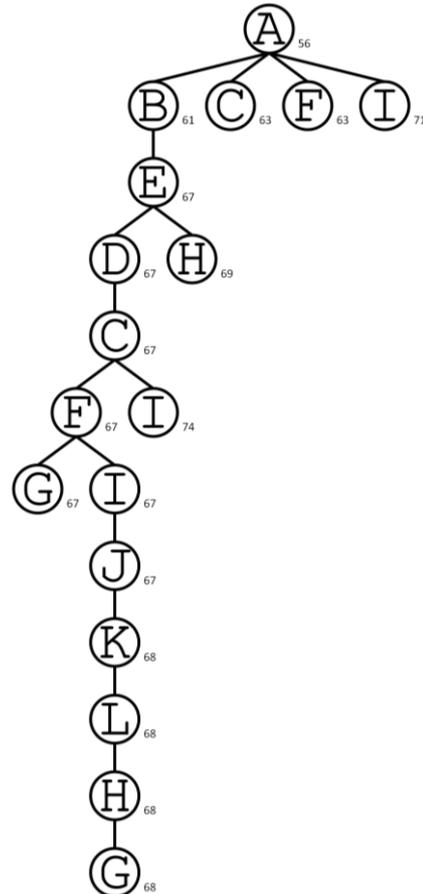
1. Simpul H; Lintasan A, B, E, D, C, F, G, H
 $r = 0$; $\hat{c}(H) = 67 + 0 + 0 = 67$
2. Simpul L; Lintasan A, B, E, D, C, F, G, H, L
 $r = 0$; $\hat{c}(L) = 67 + 0 + 0 = 67$
3. Simpul K; Lintasan A, B, E, D, C, F, G, H, L, K
 $r = 0$; $\hat{c}(K) = 67 + 0 + 0 = 67$
4. Simpul J; Lintasan A, B, E, D, C, F, G, H, L, K, J
 $r = 16$; $\hat{c}(J) = 67 + 0 + 16 = 83$
5. Simpul I; Lintasan A, B, E, D, C, F, G, H, L, K, J, I
 $r = 0$; $\hat{c}(I) = 83 + 0 + 0 = 83$



Gambar 8 Pohon Ruang Status Hasil Ekspansi Simpul G

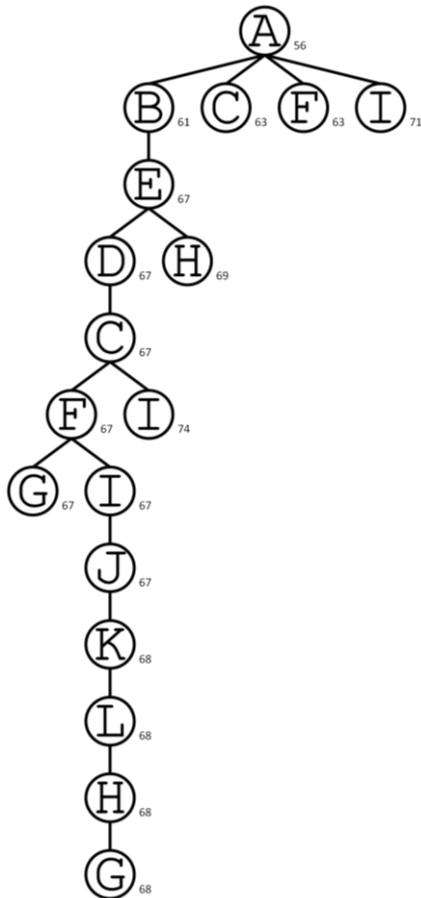
Selanjutnya, dilakukan ekspansi pada simpul I sampai selesai:

1. Simpul J; Lintasan A, B, E, D, C, F, I, J
 $r = 0$; $\hat{c}(J) = 67 + 0 + 0 = 67$
2. Simpul K; Lintasan A, B, E, D, C, F, I, J, K
 $r = 1$; $\hat{c}(K) = 67 + 0 + 1 = 68$
3. Simpul L; Lintasan A, B, E, D, C, F, I, J, K, L
 $r = 0$; $\hat{c}(L) = 68 + 0 + 0 = 68$
4. Simpul H; Lintasan A, B, E, D, C, F, I, J, K, L, H
 $r = 0$; $\hat{c}(H) = 68 + 0 + 0 = 68$
5. Simpul G; Lintasan A, B, E, D, C, F, I, J, K, L, H, G
 $r = 0$; $\hat{c}(G) = 68 + 0 + 0 = 68$



Gambar 9 Pohon Ruang Status Hasil Ekspansi Simpul I

Dari kedua ekspansi di atas, kita dapat mengetahui bahwa *cost* minimum didapat dari hasil ekspansi simpul I yaitu 68 (enam puluh delapan) seperti yang ditunjukkan pada Gambar 10 berikut ini.



Gambar 10 Pohon Ruang Status Solusi

REFERENSI

- [1] Munir, Rinaldi, "Diktat Kuliah IF3051 Strategi Algoritma", Bandung, 2009.
- [2] <http://rinaldimunir.wordpress.com/2009/02/25/nama-nama-mereka-sudah-terpatri/>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 9 Desember 2011

Chita Najmi Nabila – 13509015

IV.3 Solusi TSP pada Tur Labtek V

Dari pencarian rute terpendek pada Tur Pengenalan Labtek V Gedung Benny Subianto, dapat kita simpulkan bahwa rute TSP terbaik yang bisa ditelusuri dari awal simpul adalah melalui lintasan A, B, E, D, C, F, I, J, K, L, H, dan G. Rute optimalnya adalah Sekretariat 1 HMIF, Selasar Labtek V, Selasar TU IF, Ruang Ketua Program Studi, Area Laboratorium 1, Ruang Kuliah, Area Laboratorium 3, Dapur Labtek V, Area Laboratorium Dasar, Sekretariat 2 HMIF, Selasar MIC, dan Area Laboratorium 2 dengan panjang lintasan 68 (enam puluh delapan) satuan panjang.

V. KESIMPULAN

1. Algoritma Branch and Bound yang menggunakan skema Breadth First Search sebagai skema traversal grafnya ini banyak digunakan sebagai metode pencarian solusi Traveling Salesman Problem.
2. Pada pencarian rute terpendek pada Tur Pengenalan Labtek V Gedung Benny Subianto, rute dengan *cost* minimum adalah A – B – E – D – C – F – I – J – K – L – H – G.
3. Penyelesaian solusi TSP dapat menggunakan fungsi pembatas pada persamaan $\hat{c}(S) = \hat{c}(R) + A(i, j) + r$.