



**UTS Semester Ganjil 2016/2017**  
**KECERDASAN BUATAN (CCH3F3)**  
**Selasa, 11 Oktober 2016 pukul 10.15 – 12.15 WIB (120 menit)**  
**Tim Dosen: SUO, IOH, KNR, MDS, SYM, UNW**

= *Open Book, No-Electronic Device* =  
= *Dilarang keras bekerja sama atau mencontek! Jika dilakukan, maka Nilai Akhir adalah E!* =

**Petunjuk**

- Bacalah soal dengan seksama sebelum mengerjakan! Kerjakan dengan teliti!
- Skor maksimal adalah 100, sekaligus menentukan nilai UTS Anda.
- Dilarang saling meminjam buku atau catatan! Untuk mengerjakan, gunakan pulpen hitam/biru.
- Berdo'alah sebelum mengerjakan ujian. ☺

<b>Nama Mahasiswa:</b> .....	<b>NIM:</b> .....	<b>Kls:</b> IF-.....	<b>Ruang:</b> .....	<b>Nilai (Diisi Dosen):</b>
---------------------------------	----------------------	-------------------------	------------------------	-----------------------------

**Salinlah pernyataan berikut:**

*Saya mengerjakan ujian ini dengan jujur dan mandiri. Jika saya melakukan pelanggaran, maka saya bersedia menerima sanksi.*

**Tanda Tangan Mahasiswa:**

.....  
.....  
.....

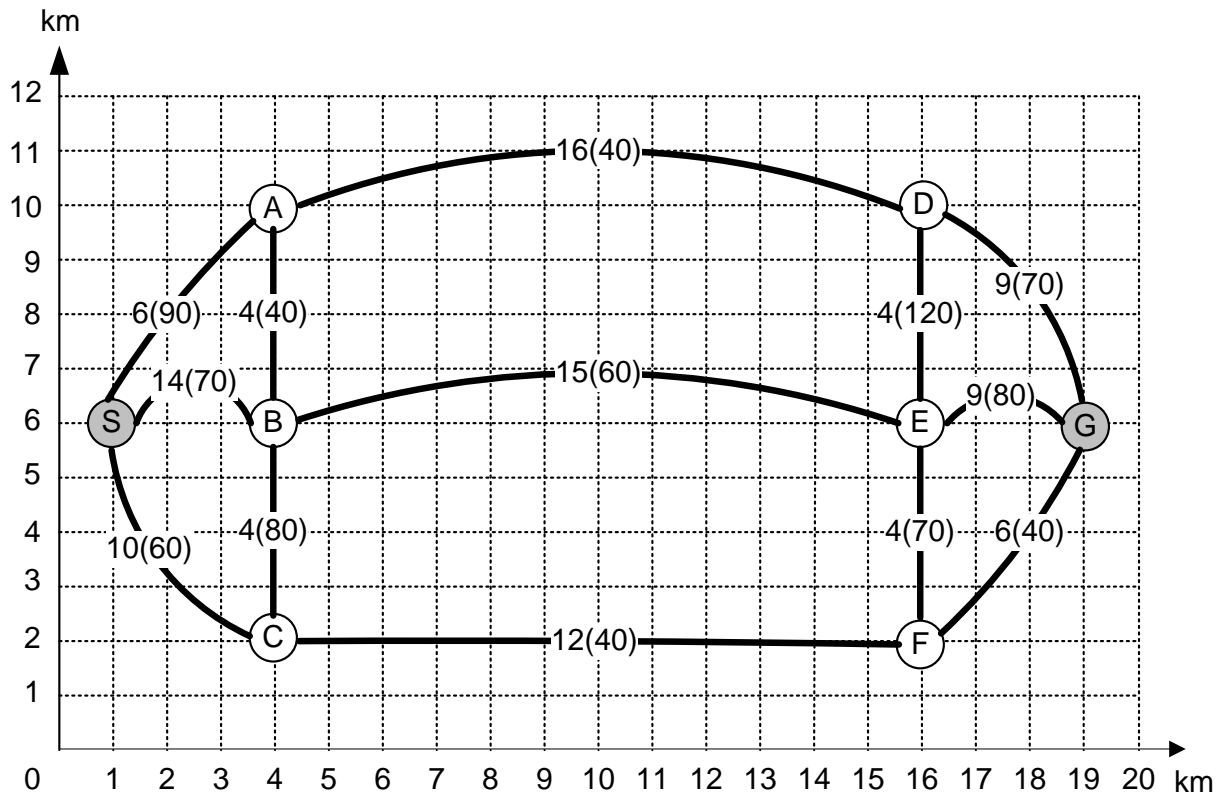
.....

Kompetensi yang ingin dicapai: Mahasiswa mampu memahami dan menerapkan beberapa metode pencarian yang termasuk ke dalam *blind search*, *heuristic search*, dan *meta-heuristic search*.

**Pencarian Rute Tercepat**

Gambar di bawah ini adalah sebuah *graph* simetris tak berarah yang menggambarkan kondisi jalan raya di suatu kota. Terdapat 8 simpul yang menyatakan persimpangan jalan dengan posisi-posisi koordinat dua dimensi (x,y). Setiap busur memiliki dua atribut: angka pertama menyatakan panjang jalan sebenarnya (dalam satuan km) dan angka kedua yang berada di dalam tanda kurung ‘(...)’ menyatakan kecepatan maksimum yang mungkin dicapai oleh setiap mobil yang melalui jalan tersebut (dalam satuan km/jam). Seorang anak baik bernama Hasan ingin memacu mobil listriknya (yang kecepatan maksimumnya 60 km/jam) secepat mungkin dari S ke G.

- a. Lakukan penelusuran algoritma *Uniform Cost Search* (UCS) secara singkat, menggunakan gambar dan penjelasan secukupnya (tidak perlu detail menampilkan setiap variabel dalam algoritma), untuk menemukan rute jalan tercepat dari S ke G beserta waktu tempuhnya yang harus dilalui Hasan. **(30 poin)**
- b. Lakukan penelusuran algoritma A\* secara singkat untuk menemukan rute jalan tercepat dari S ke G beserta waktu tempuhnya yang harus dilalui Hasan. **(30 poin)**
- c. Rancanglah kromosom, fungsi *fitness*, rekombinasi, dan mutasi yang tepat sehingga Algoritma Genetika dapat secara efisien menemukan rute tercepat dari S ke G beserta waktu tempuhnya yang harus dilalui Hasan. **(40 poin)**

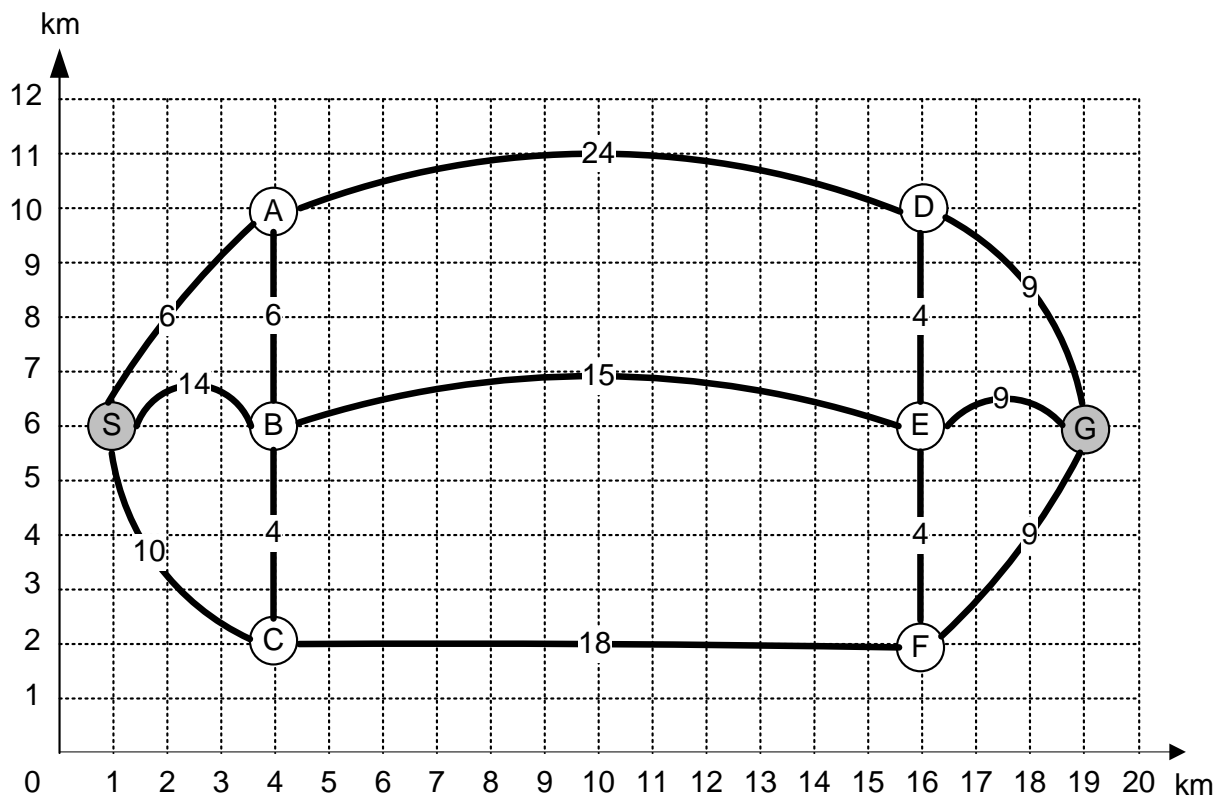
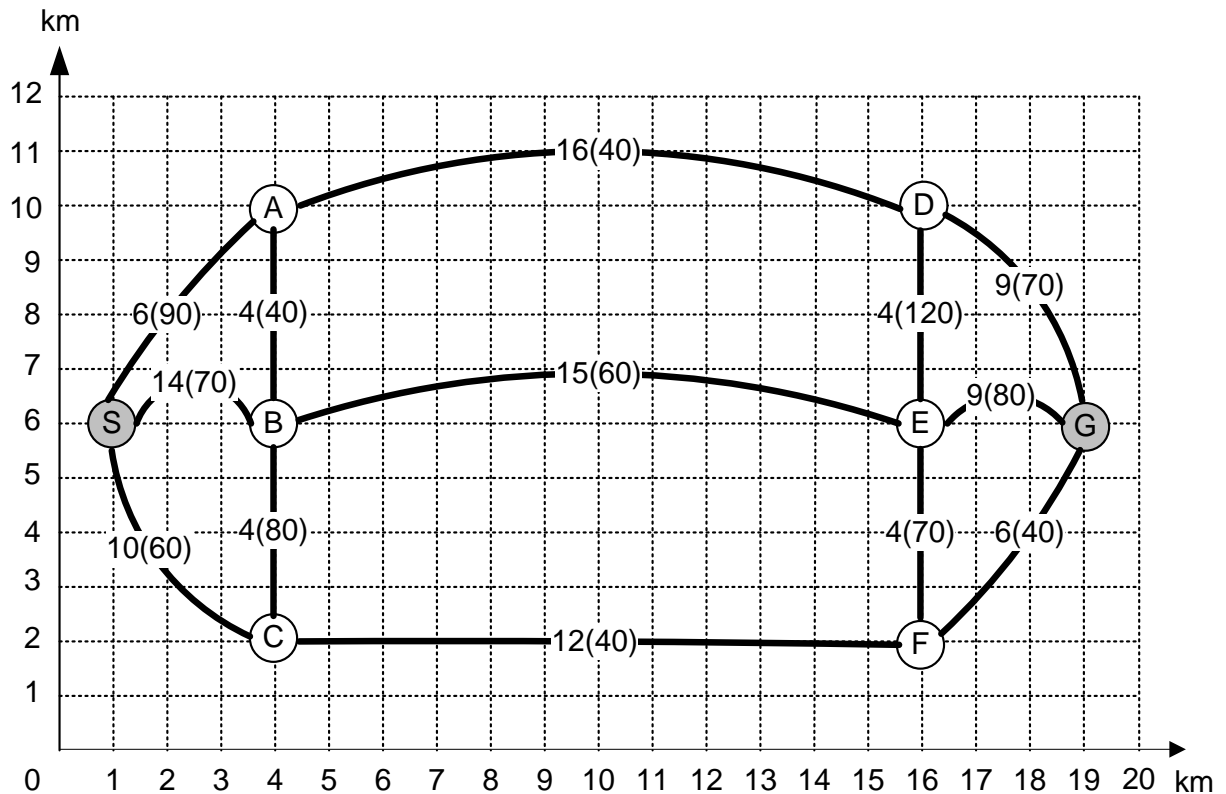


#### **Petunjuk teknis:**

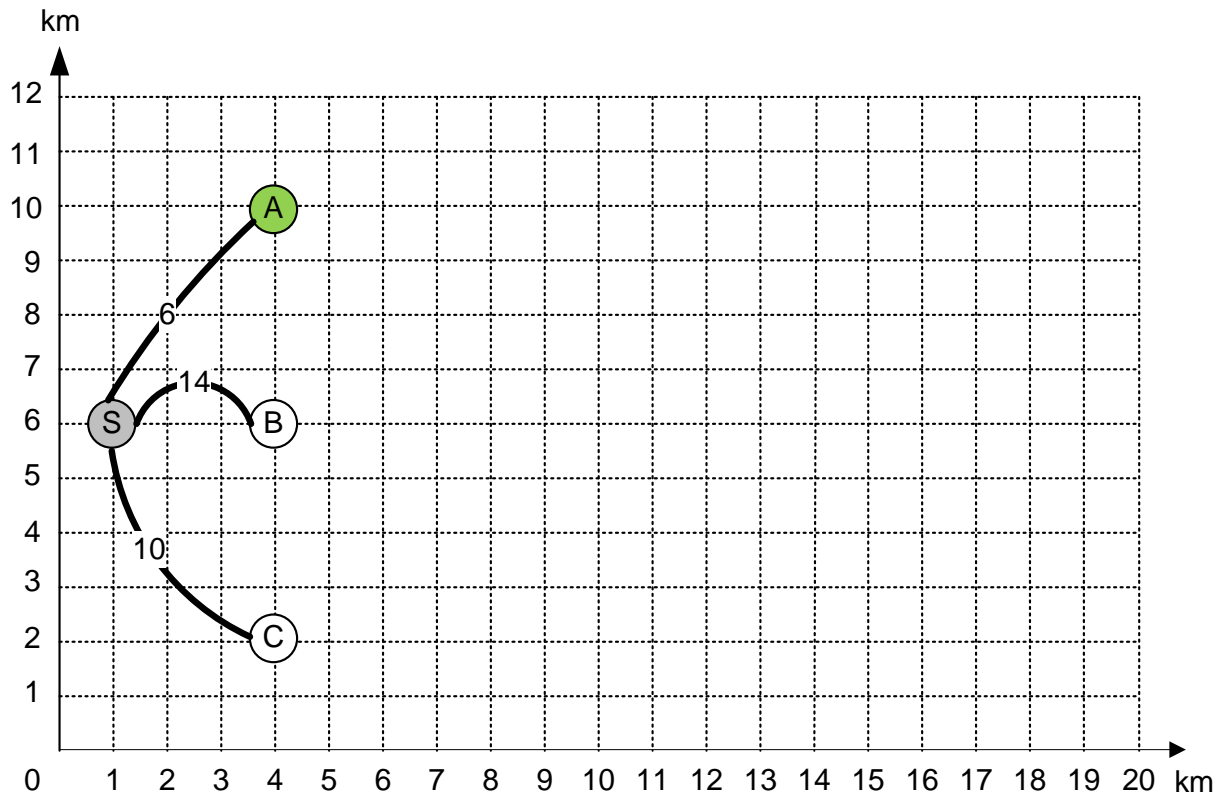
- Hasan selalu memacu mobilnya dengan kecepatan maksimum yang dapat dicapai mobil listriknya. Pada jalan tol, di mana mobil-mobil lain bisa mencapai kecepatan 100 km/jam, Hasan hanya bisa berjalan 60 km/jam. Pada jalanan sempit dan macet, di mana semua mobil hanya bisa berjalan dengan kecepatan 40 km/jam, maka mobil Hasan juga hanya bisa berjalan 40 km/jam (walaupun sebenarnya dapat dipacu hingga 60 km/jam).
- Waktu tempuh  $t$  dihitung menggunakan rumus  $t = \frac{d}{v}$ , di mana  $d$  jarak dan  $v$  kecepatan.
- Pada pencarian rute terpendek (bukan rute tercepat), *heuristic* diperoleh dari jarak garis lurus antara setiap titik ke titik tujuan (G) yang dihitung menggunakan rumus  $\sqrt{\Delta x^2 + \Delta y^2}$ , di mana  $\Delta x$  adalah selisih dua titik dalam sumbu  $x$  dan  $\Delta y$  adalah selisih dua titik dalam sumbu  $y$ .
- Untuk pencarian rute tercepat, berhati-hatilah dalam menghitung *heuristic* waktu tempuh. Pastikan bahwa biaya perkiraan (*heuristic cost*) selalu lebih kecil atau sama dengan biaya sebenarnya (*real cost*).

## Solusi

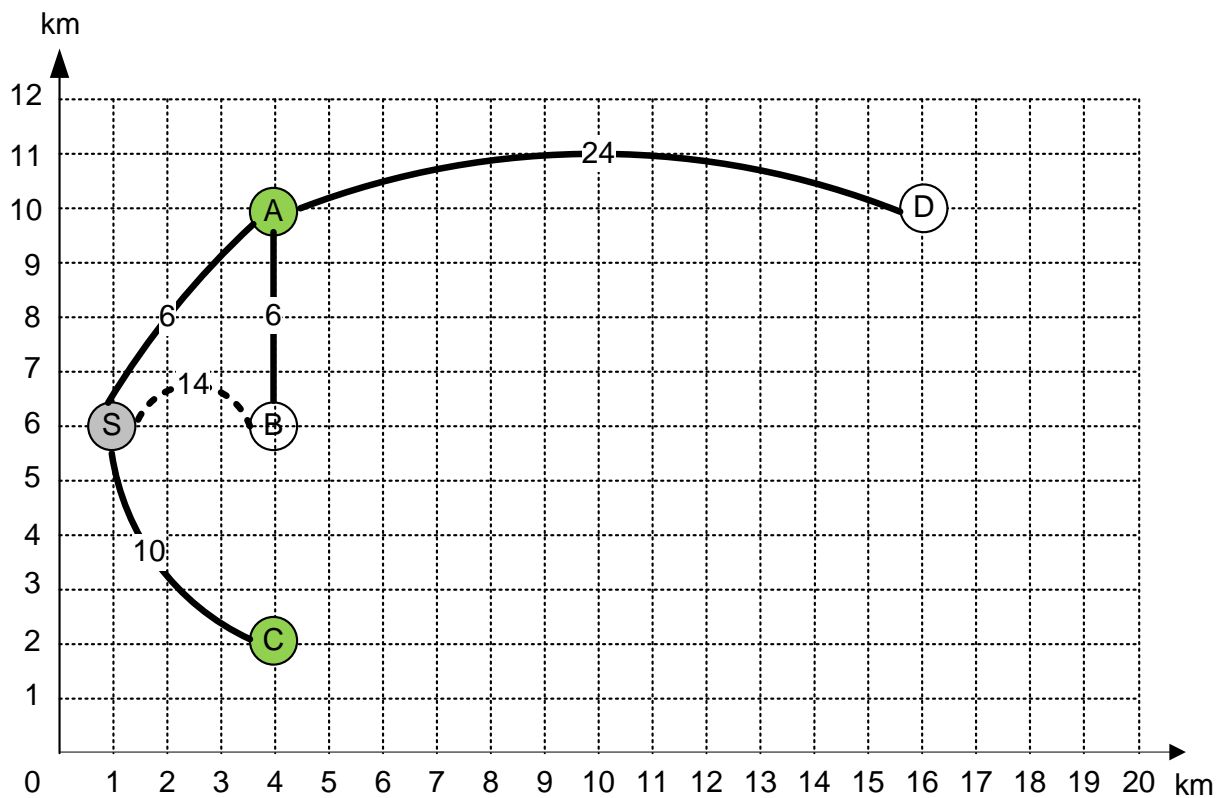
- a. Untuk menyelesaikan kasus tersebut menggunakan algoritma UCS, hal pertama yang harus dilakukan adalah mengubah nilai-nilai busur pada graph tersebut dari jarak (km) menjadi waktu (dalam hal ini lebih mudah menggunakan menit (bukan jam) karena kecepatan mobil adalah 60km/jam atau 60 km per 60 menit, yang berarti waktu tempuh 1 km adalah 1 menit).



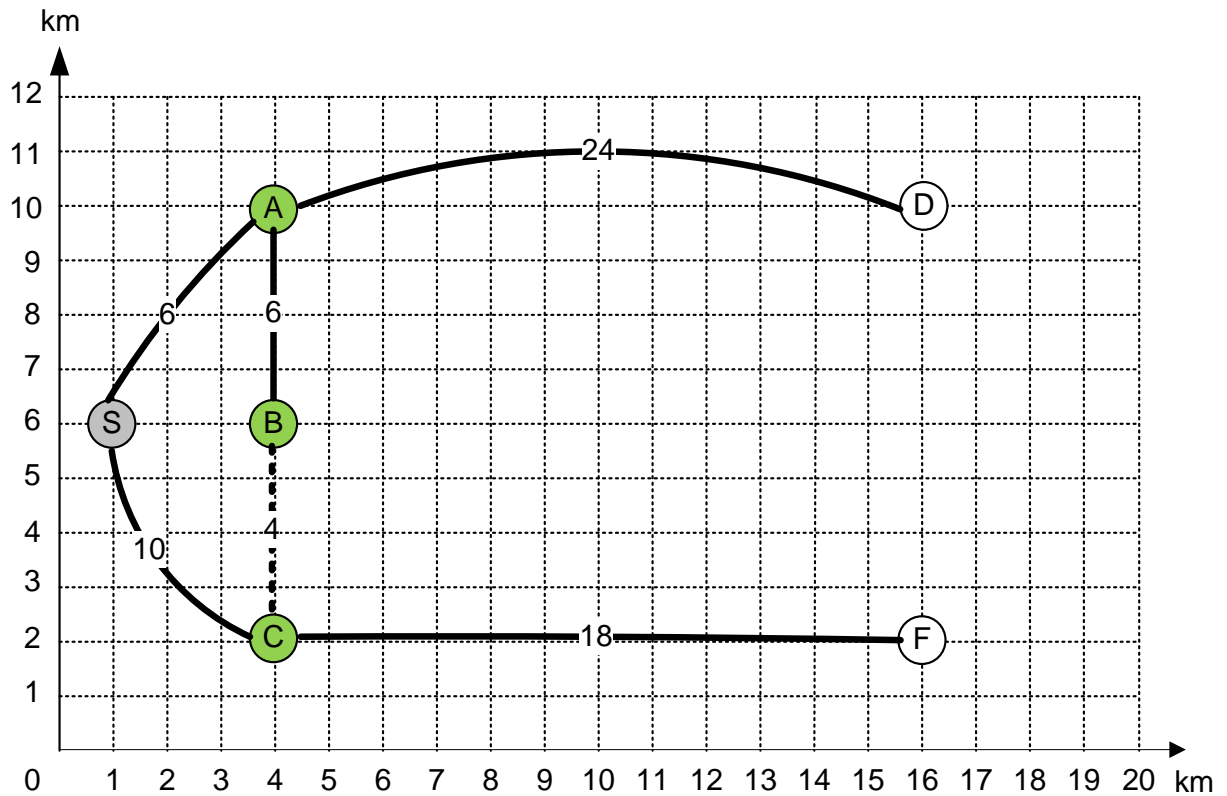
Selanjutnya, penelusuran menggunakan UCS dapat dilakukan dengan cara sebagai berikut:



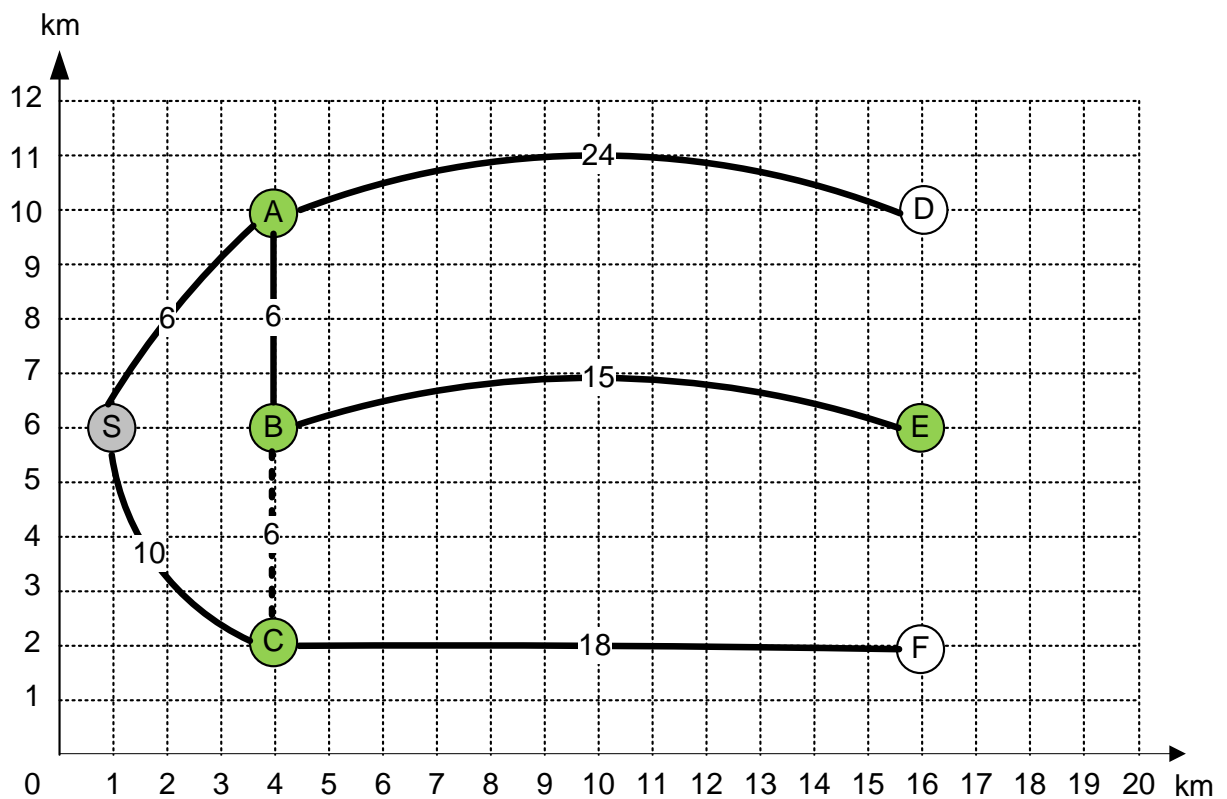
Dari S dibangkitkan A, B, dan C. Simpul A dengan waktu 6 menit terpilih sebagai *best node*.



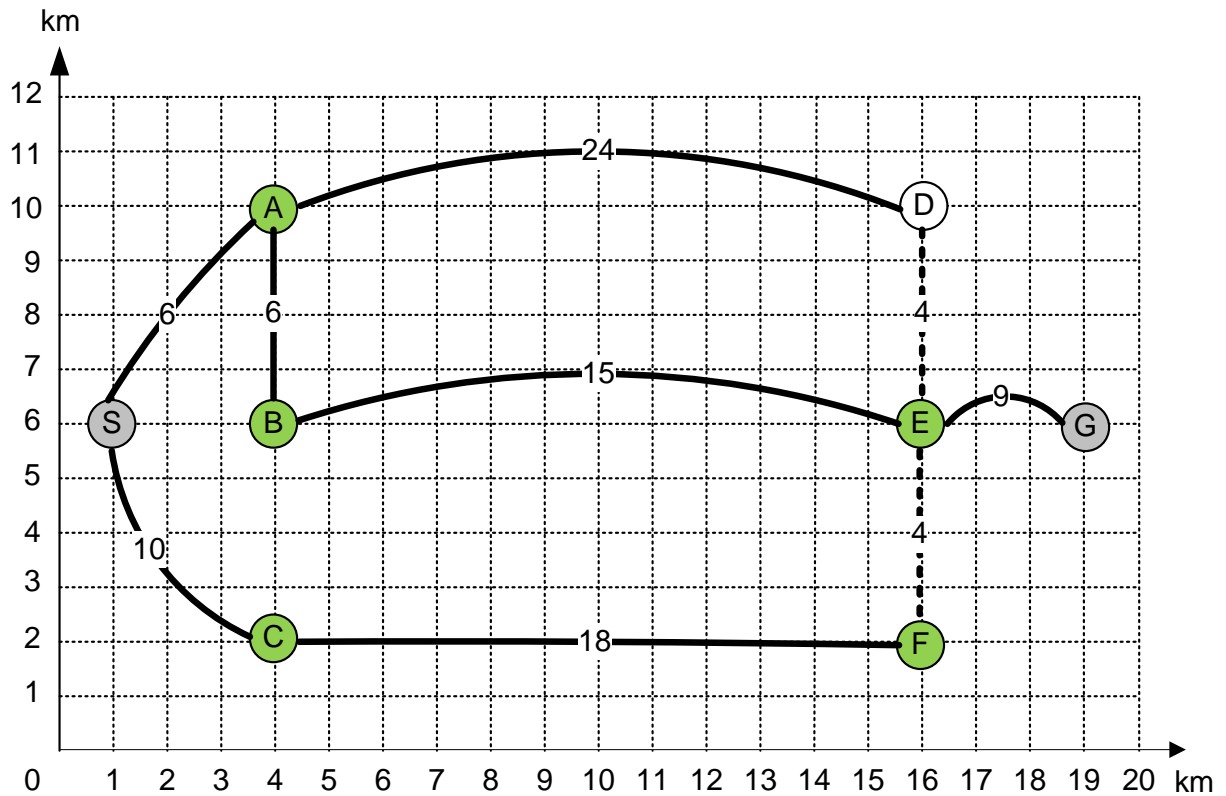
Dari A dibangkitkan B dan D. Parent dari B diubah dari S menjadi A karena waktu tempuhnya lebih kecil. Simpul C dengan waktu tempuh 10 menit terpilih sebagai *best node*.



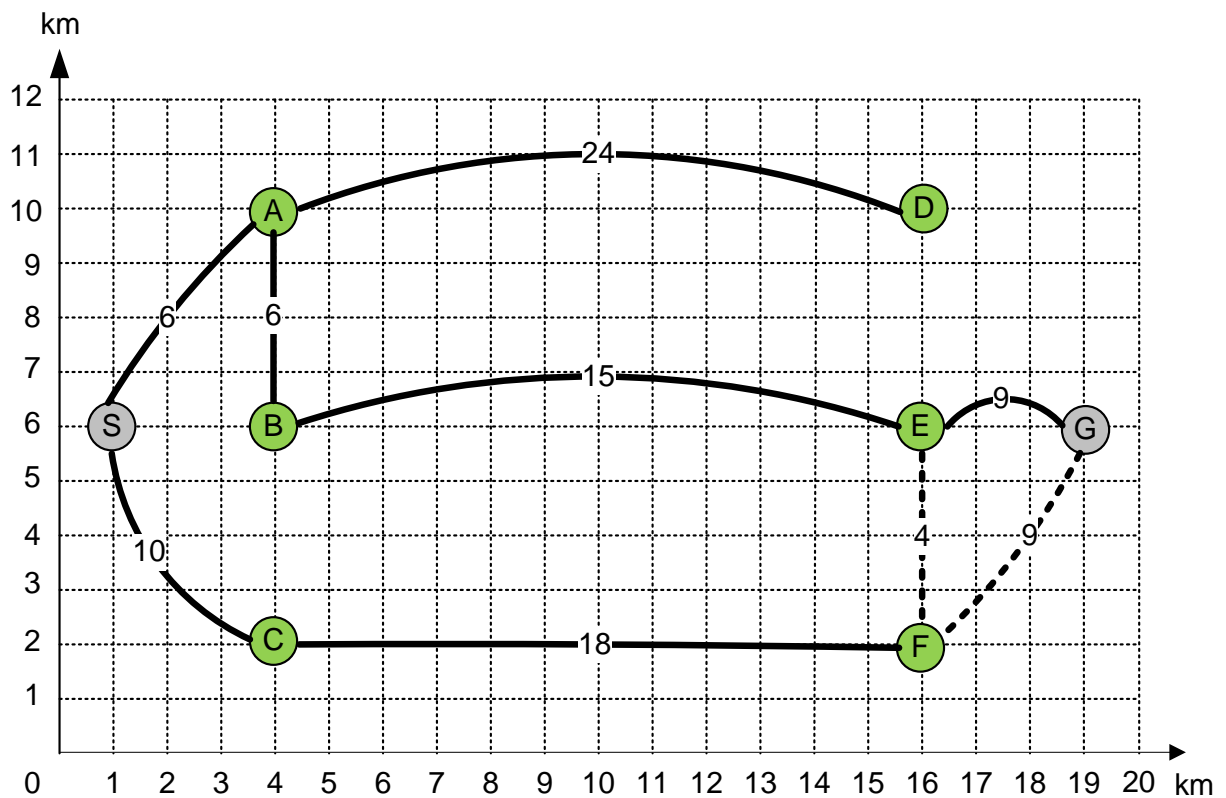
Dari C dibangkitkan B dan F. Parent dari B tetap A (tidak perlu diubah ke C) karena waktu tempuhnya lebih kecil. Simpul B dengan waktu tempuh 12 menit terpilih sebagai *best node*.



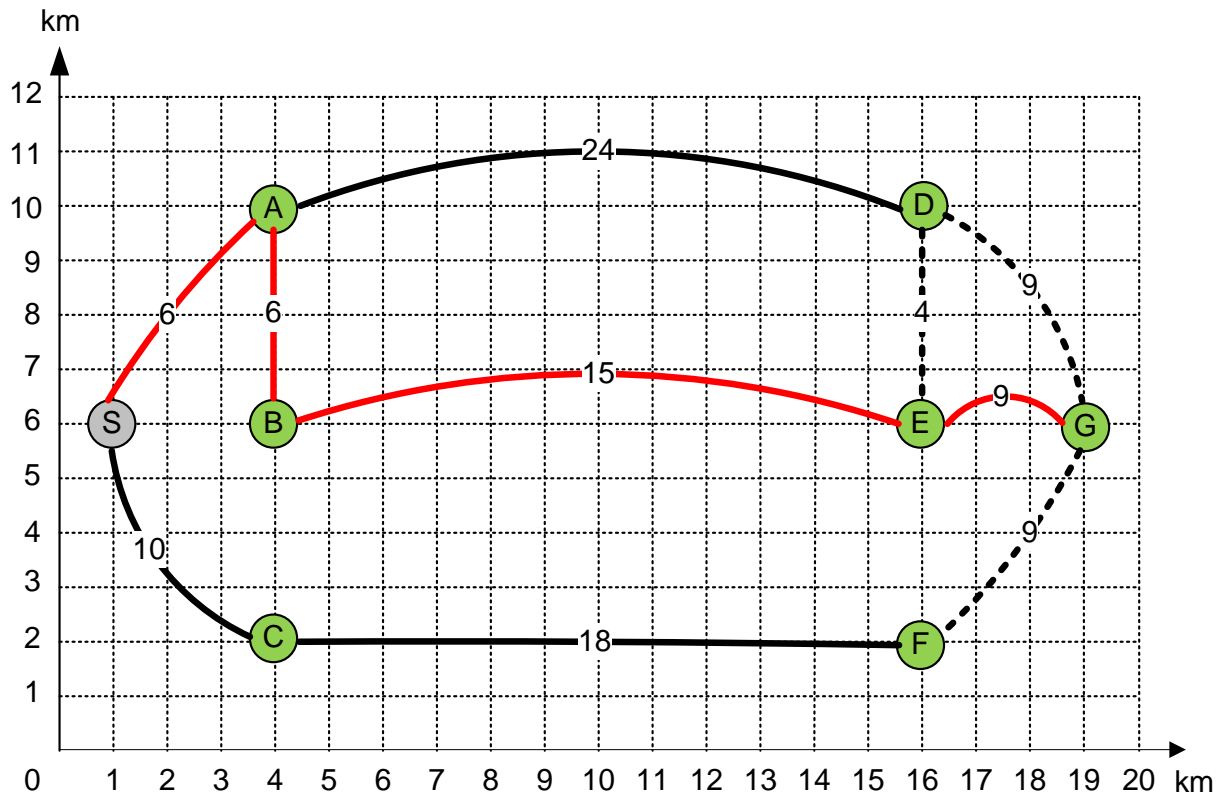
Dari B dibangkitkan C dan E. Parent dari C tetap S (tidak perlu diubah ke B) karena waktu tempuhnya lebih kecil. Simpul E dengan waktu tempuh 27 menit terpilih sebagai *best node*.



Dari E dibangkitkan D, F, dan G. Parent dari D tetap A (tidak perlu diubah ke E) dan parent dari F tetap C (tidak perlu diubah ke E). Walaupun simpul G sudah dibangkitkan, penelusuran belum dihentikan karena G belum menjadi *best node*. Simpul F dengan waktu tempuh 28 menit terpilih sebagai *best node*.



Dari F dibangkitkan E dan G. Parent dari E tetap B (tidak perlu diubah ke F) dan parent dari G tetap E (tidak perlu diubah ke F). Walaupun simpul G sudah dibangkitkan, penelusuran belum dihentikan karena G belum menjadi *best node*. Simpul D dengan waktu tempuh 30 menit terpilih sebagai *best node*.

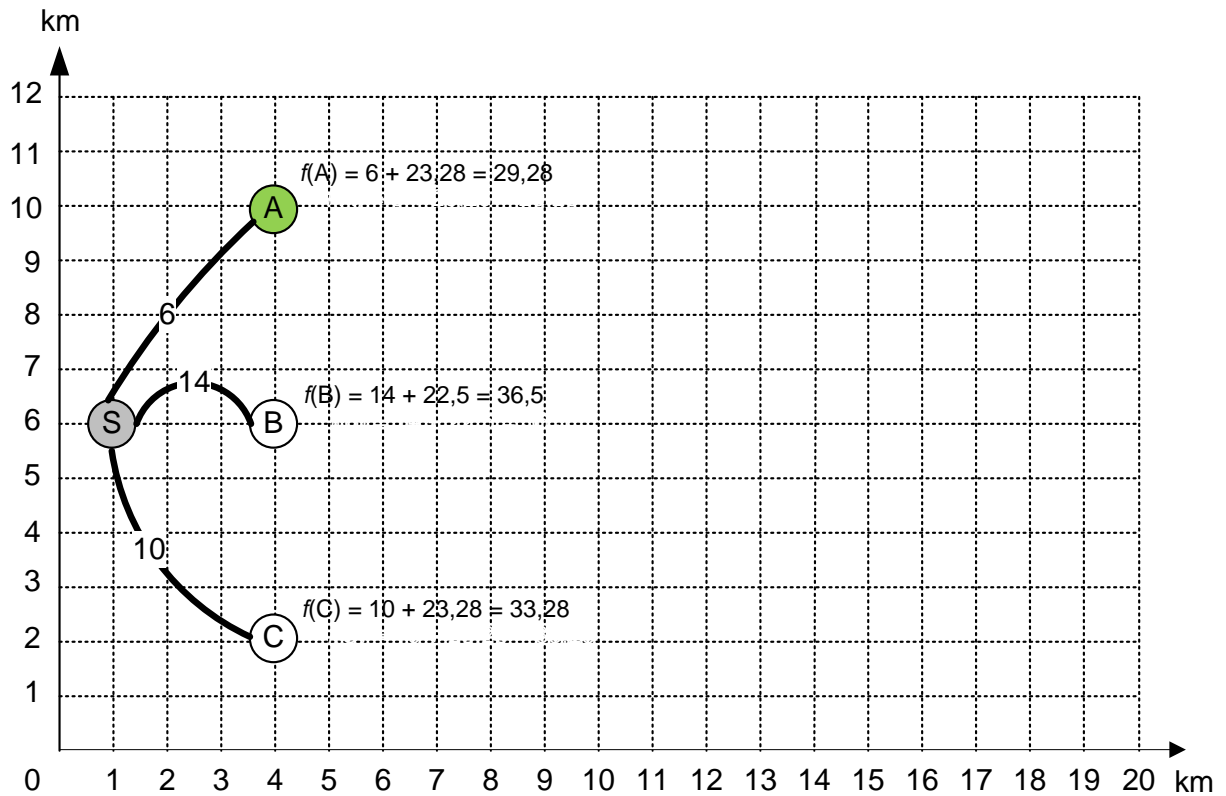


Dari D dibangkitkan E dan G. Parent dari E tetap B (tidak perlu diubah ke D) dan parent dari G tetap E (tidak perlu diubah ke D). Simpul G (melalui E) dengan waktu tempuh 36 menit terpilih sebagai *best node*. Dengan demikian, penelusuran dapat dihentikan dan menghasilkan rute tercepat S-A-B-E-G dengan total waktu tempuh = 6 + 6 + 15 + 9 = **36 menit**.

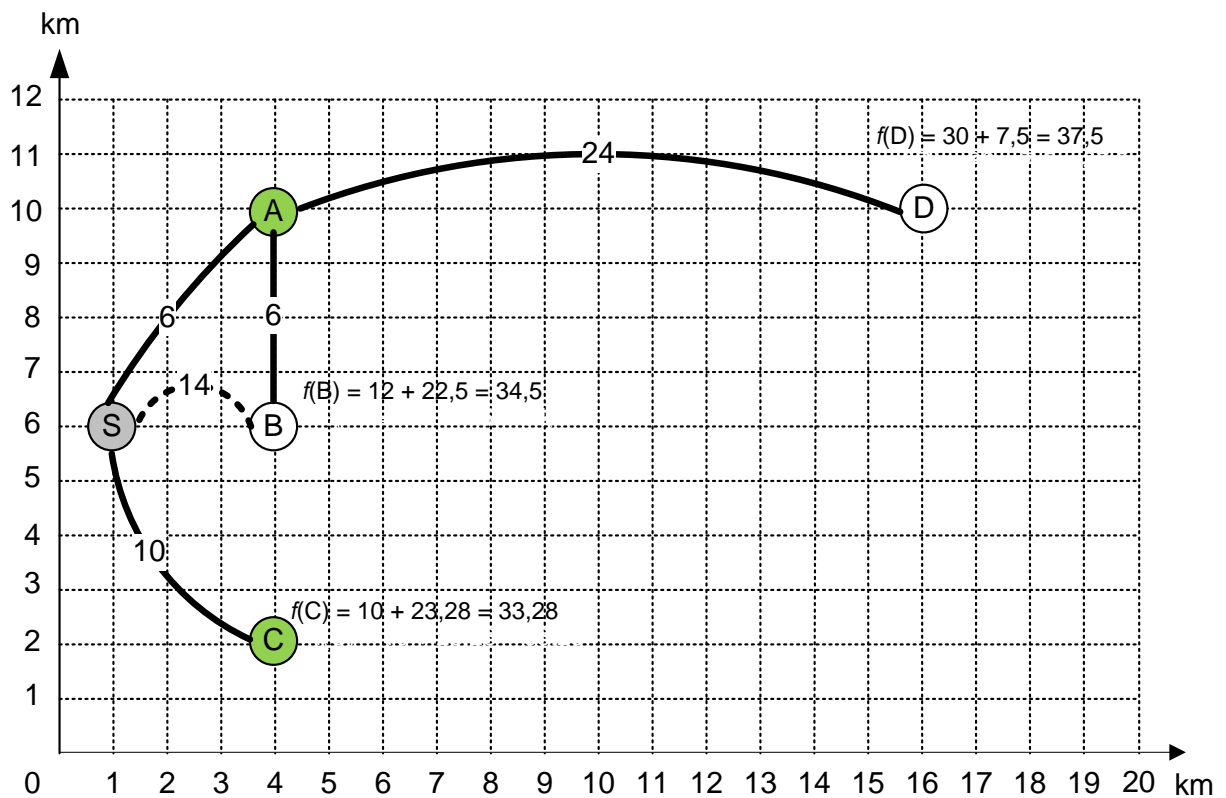
- b. Untuk menyelesaikan kasus tersebut menggunakan algoritma A\*, hal pertama yang harus dilakukan adalah mengubah nilai-nilai busur pada graph tersebut dari jarak (km) menjadi waktu (dalam menit) seperti pada poin (a) di atas. Selanjutnya, definisikan nilai-nilai heuristik yang berupa estimasi waktu tempuh (juga dalam menit) untuk setiap simpul yang ada. Mengingat nilai heuristik  $\leq$  nilai sesungguhnya, maka kecepatan mobil yang digunakan harus sama dengan kecepatan minimal yang ada pada peta jalan raya tersebut, yaitu 40 km/jam (40 km per 60 menit). Nilai heuristik (waktu tempuh estimasi) dari simpul  $n$  ke  $G$  dihitung menggunakan rumus  $h(n) = \frac{d}{v}$ , di mana  $d$  adalah jarak garis lurus dari simpul  $n$  ke  $G$  (dalam km) dan  $v = 40/60 = 4/6$  km/menit.

$n$	S	A	B	C	D	E	F	G
$h(n)$	$18 * 6/4 = 27$ menit	$15,52 * 6/4 = 23,28$	$15 * 6/4 = 22,5$ menit	$15,52 * 6/4 = 23,28$	$5 * 6/4 = 7,5$ menit	$3 * 6/4 = 4,5$ menit	$5 * 6/4 = 7,5$ menit	$0 * 6/4 = 0$ menit

Selanjutnya, penelusuran menggunakan A\* dapat dilakukan dengan cara sebagai berikut:

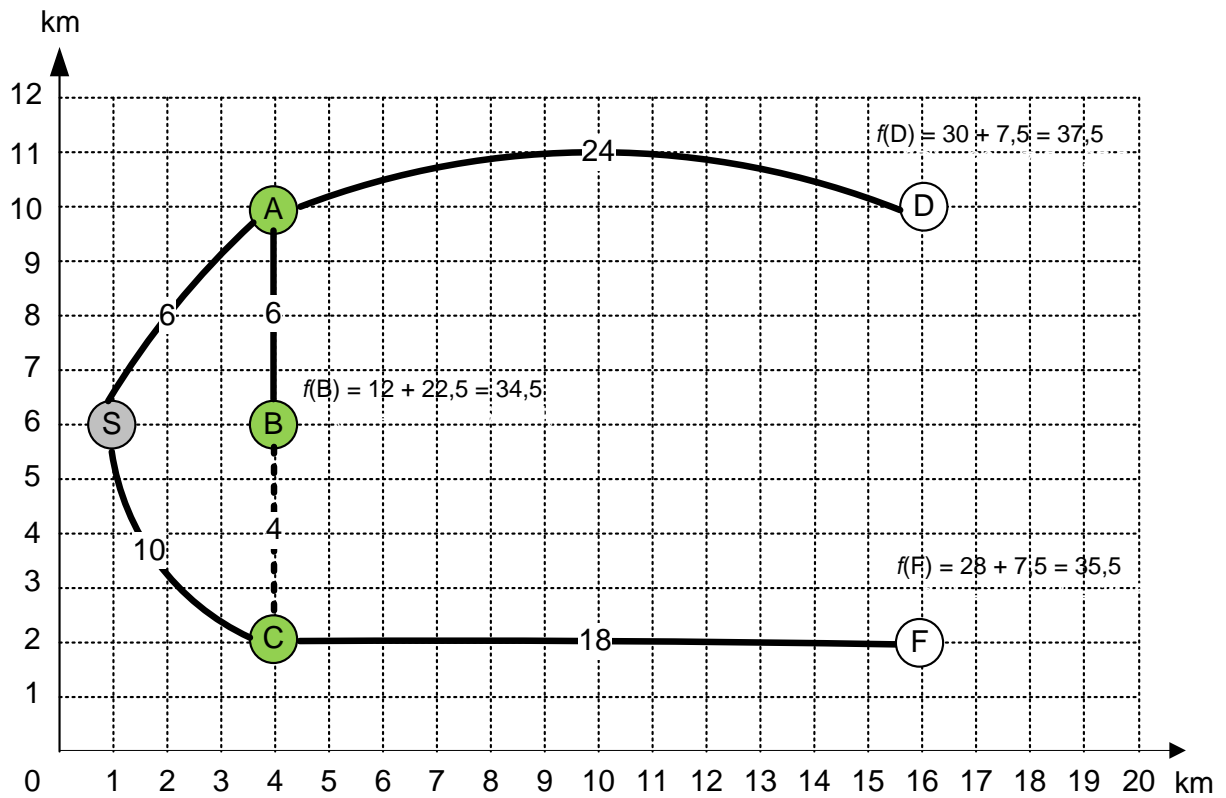


Dari S dibangkitkan A, B, dan C. Simpul A dengan jumlah waktu tempuh sesungguhnya dan waktu tempuh estimasi  $f(A) = g(A) + h(A) = 6 + 23,28 = 29,28$  menit terpilih sebagai *best node*.

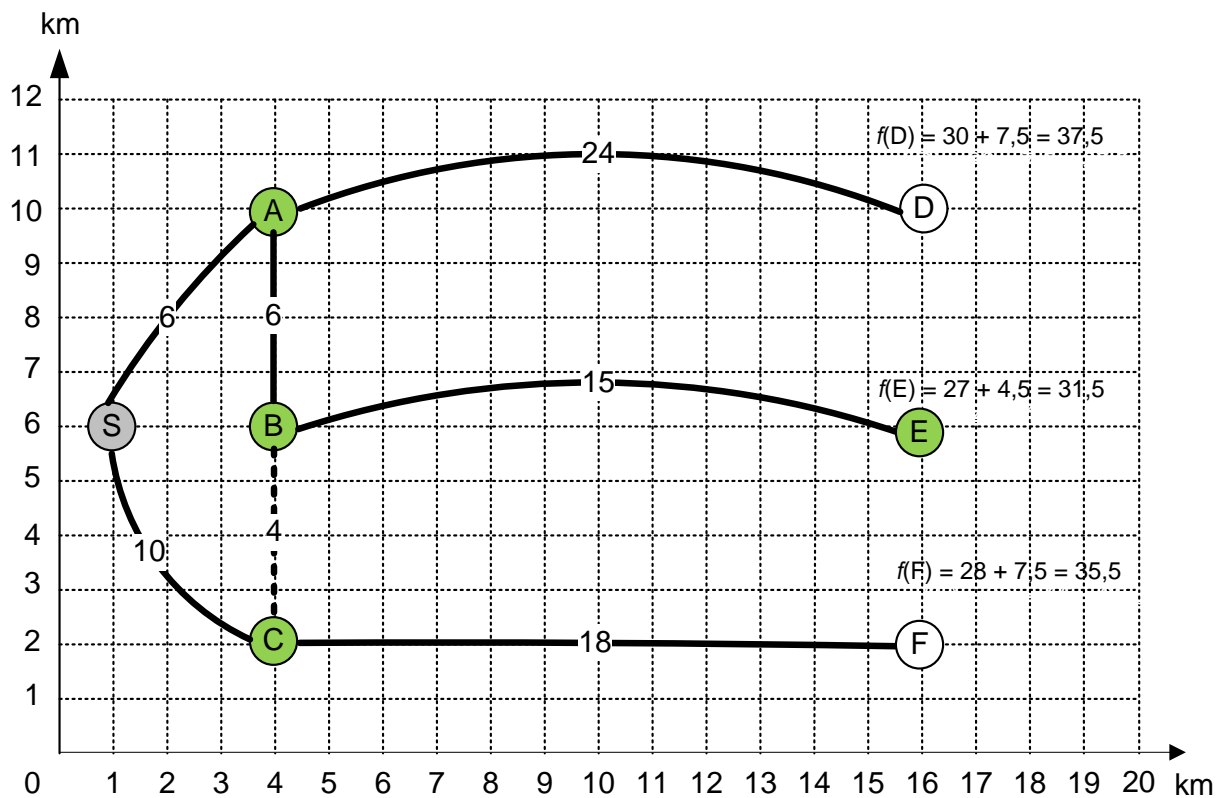


Dari A dibangkitkan B dan D. Parent dari B diubah dari S ke A dan  $f(B)$  diubah dari 36,5 menjadi 34,5. Simpul C dengan jumlah waktu tempuh sesungguhnya dan waktu tempuh estimasi  $f(C) = g(C) + h(C) = 10 + 23,28 = 33,28$  menit terpilih sebagai *best node*.

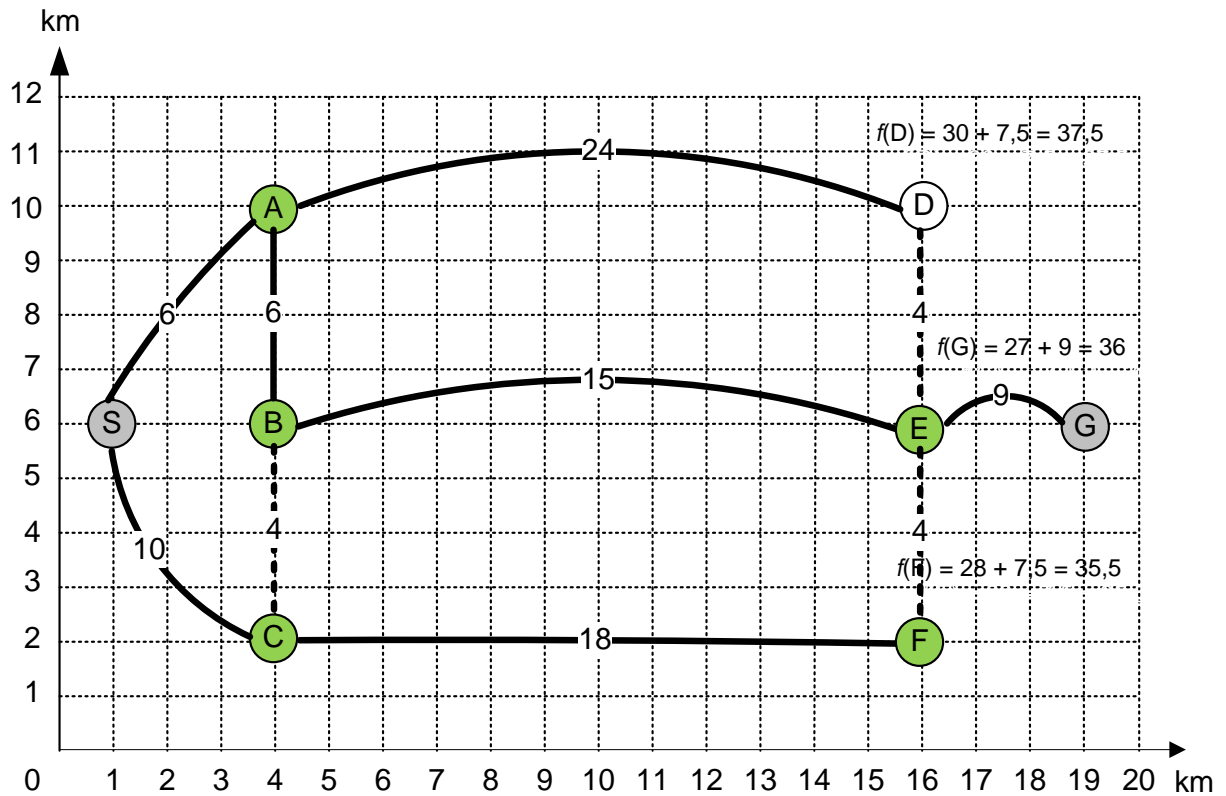




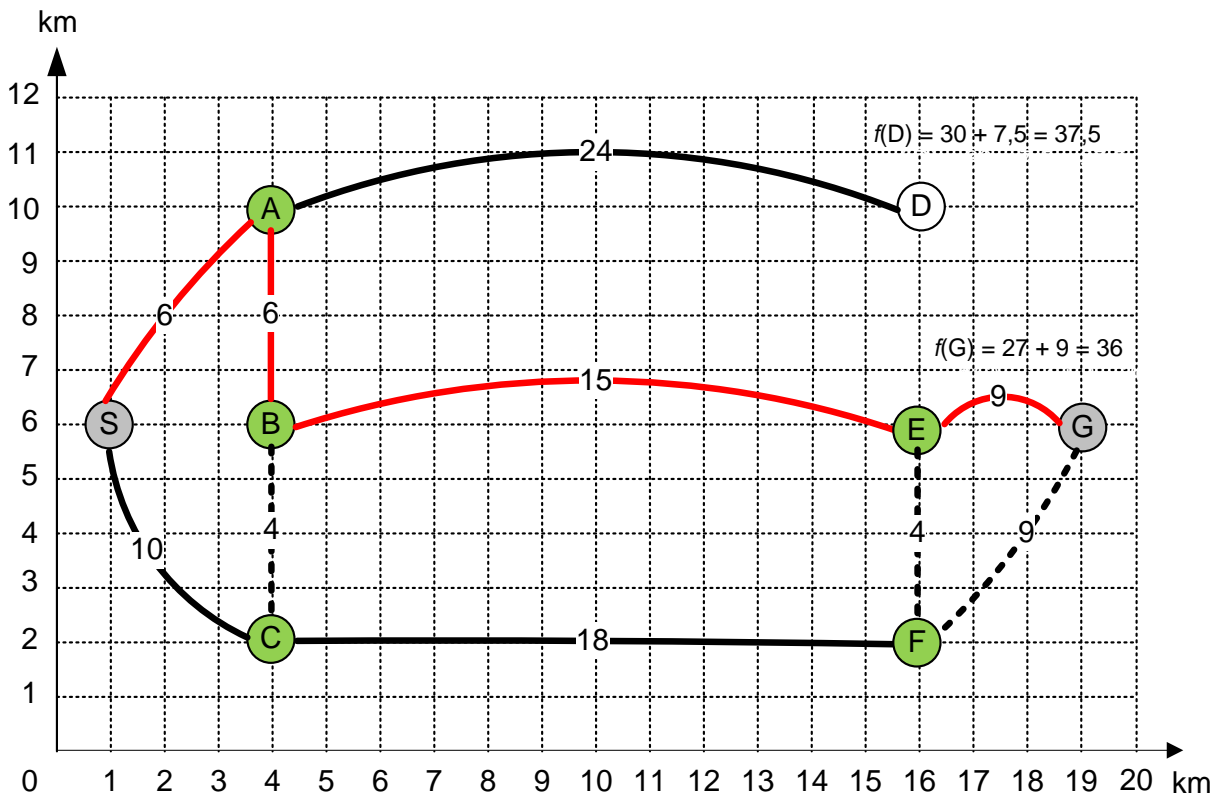
Dari C dibangkitkan B dan F. Parent dari B tidak perlu diubah dari A ke C. Simpul B dengan jumlah waktu tempuh  $f(B) = 34.5$  menit terpilih sebagai *best node*.



Dari B dibangkitkan C dan E. Parent dari C tidak perlu diubah dari S ke B. Simpul E dengan jumlah waktu tempuh  $f(E) = 31.5$  menit terpilih sebagai *best node*.



Dari E dibangkitkan D, F, dan G. Parent dari D tidak perlu diubah dari A ke E. Parent dari F tidak perlu diubah dari C ke E. Simpul F dengan jumlah waktu tempuh  $f(F) = 35,5$  menit terpilih sebagai *best node*. Walaupun simpul G sudah dibangkitkan, namun penelusuran belum berhenti karena G belum terpilih sebagai *best node*.



Dari F dibangkitkan E dan G. Parent dari E tidak perlu diubah dari B ke F. Simpul G dengan jumlah waktu tempuh  $f(G) = 36$  menit terpilih sebagai *best node*. Dengan demikian, penelusuran belum dihentikan dan proses *backtracking* menghasilkan rute tercepat **S-A-B-E-G** dengan total waktu tempuh **36 menit**.

- c. Untuk menyelesaikan kasus tersebut menggunakan algoritma genetika, dua hal terpenting yang harus diperhatikan adalah: 1) bagaimana menjamin kromosom selalu menghasilkan calon solusi yang valid walaupun dilakukan operasi pindah silang dan mutasi; 2) bagaimana menjamin fungsi *fitness* dapat membuat evolusi terus terjadi (tidak mengalami stagnasi).

### Desain Kromosom

Ada banyak representasi kromosom yang bisa digunakan, di antaranya adaah sebagai berikut:

#### Representasi 1

A	C	B	F	E	D
1	2	3	4	5	6

Nilai gen berupa abjad/huruf yang menyatakan simpul (persimpangan jalan) yang berjumlah enam (S dan G tidak perlu direpresentasikan ke dalam kromosom karena sudah pasti dikunjungi) sedangkan indeks menyatakan urutan kunjungan. Jika dari simpul sekarang ke simpul  $n$  tidak ada busur (koneksi), maka simpul  $n$  disimpan ke sebuah array OPEN secara berurutan. Jadi, kromosom di atas menyatakan calon solusi berupa rute **S-A-B-E-G** dengan OPEN = [D, F]. Bagaimana jika penelusuran gen tersebut tidak menghasilkan rute yang valid? Mudah saja, ambil simpul yang ada di OPEN untuk membangun rute valid. Sebagai contoh, pada kromosom di bawah ini

A	E	F	B	D	C
1	2	3	4	5	6

penelusuran pertama menghasilkan rute tidak valid **S-A-B-C** yang tidak sampai ke simpul G dengan OPEN = [E, F, D]. Namun, rute tersebut dapat diteruskan dengan mengambil simpul F pada OPEN sehingga dihasilkan rute valid **S-A-B-C-F-G** dengan OPEN = [E, D]. Cara lain yang lebih sederhana (mungkin saja tanpa penggunaan array OPEN) adalah dengan memperpanjang kromosom tersebut menjadi (misalnya lima kali lipat) dari semula, seperti pada gambar berikut:

A	F	F	A	E	B	E	...	B	D	F	A
1	2	3	4	5	6	7		27	28	29	30

Jika sebuah simpul muncul lebih dari satu kali, maka diambil yang pertama. Jika penelusuran 30 gen di atas tidak menghasilkan rute valid (meskipun kemungkinannya sangat kecil), maka dapat diacak simpul sembarang atau menggunakan array OPEN seperti cara sebelumnya.

#### Representasi 2

1	1	0	0	1	0
A	B	C	D	E	F

Nilai gen berupa angka biner yang menyatakan dikunjungi/tidak dan jumlahnya juga sama dengan enam (S dan G tidak perlu direpresentasikan) sedangkan indeks menyatakan simpul. Jadi, kromosom pada gambar di atas menyatakan sebuah calon solusi berupa rute **S-A-B-E-G**. Bagaimana jika penelusuran gen tersebut tidak menghasilkan rute yang valid? Mudah saja, ambil simpul yang bernilai 0 (secara berurutan) untuk membangun rute valid. Sebagai contoh, pada kromosom di bawah ini

1	1	0	1	0	1
A	B	C	D	E	F

penelusuran pertama menghasilkan rute tidak valid **S-A-B-** yang tidak sampai ke simpul G. Namun, rute tersebut dapat diteruskan dengan mengambil simpul bernilai 0 sehingga dihasilkan rute valid **S-A-B-C-F-G**. Cara lain yang lebih sederhana adalah dengan memperpanjang kromosom tersebut menjadi (misalnya lima kali lipat) dari semula, seperti pada representasinya 1 di atas.

### Fungsi Fitness

Ada banyak model matematis untuk membangun fungsi fitness, di antaranya adaah sebagai berikut:

$$f = \frac{1}{h}$$

di mana  $h$  adalah total waktu tempuh yang dihasilkan oleh rute pada suatu kromosom. Konstanta  $a$  tidak diperlukan karena  $h$  tidak mungkin bernilai 0. Cara lainnya adalah dengan model matematis

$$f = a - h$$

di mana  $h$  adalah total waktu tempuh yang dihasilkan oleh rute pada suatu kromosom dan  $a$  adalah bilangan yang lebih besar dari  $h$ .

### Rekombinasi

Metode rekombinasi harus didesain sesuai dengan representasi kromosom yang digunakan. Jika menggunakan representasi 1 di atas (tanpa memperpanjang kromosom), maka rekombinasi harus menggunakan model rekombinasi untuk representasi permutasi, yaitu: *order crossover*, *partially mapped crossover*, *cycle crossover*, atau *edge recombination*. Namun jika menggunakan representasi 1 dengan memperpanjang kromosom lima kali lipat, maka rekombinasi satu titik potong dapat digunakan. Jika menggunakan representasi 2 (tanpa atau dengan memperpanjang kromosom), rekombinasi satu titik potong dapat digunakan.

### Mutasi

Metode mutasi harus didesain sesuai dengan representasi kromosom yang digunakan. Jika menggunakan representasi 1 di atas (tanpa memperpanjang kromosom), maka mutasi harus menggunakan model mutasi untuk representasi permutasi, yaitu: *Mutasi pertukaran (Swap Mutation)*, *Mutasi penyisipan (Insert Mutation)*, *Mutasi pengacakan (Scramble Mutation)*, *Mutasi pembalikan (Inversion Mutation)*. Namun jika menggunakan representasi 1 dengan memperpanjang kromosom lima kali lipat, maka mutasi dengan pengacakan nilai gen dapat digunakan, misal nilai gen 'A' dimutasi menjadi sembarang abjad di antara 'B' dan 'F'. Jika menggunakan representasi 2 (tanpa atau dengan memperpanjang kromosom), mutasi dengan pengacakan nilai gen dapat digunakan.