Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Ujian Tengah Semester **IF2211 Strategi Algoritma**
Jumat, 8 Maret 2019
Waktu: 120 menit
Dosen: Masayu Leylia Khodra, Nur Ulfa Maulidevi, Rinaldi Munir

---

Berdoalah terlebih dahulu agar Anda sukses dalam ujian ini!

Brute Force + Divide and Conquer

1. (*Inversion problem*) *Netflix* menggunakan sistem rekomendasi untuk merekomendasikan film yang anda sukai. *Netflix* mencoba mencocokkan film kesukaanmu dengan film lainnya. Sistem rekomendasi tersebut adalah sbb: Misalkan kamu me-rangking *n* buah film. Selanjutnya, *Netflix* memeriksa basisdatanya untuk mencari orang dengan kesukaan film yang mirip. Ukuran kemiripan yang digunakan adalah jumlah inversi antara kedua rangking. Misalkan ranking dari orang tersebut adalah 1, 2, 3, …, *n*, sedangkan rangking dari kamu adalah a_1, a_2, …, a_n. Film *i* dan film *j* disebut inversi jika $i < j$ tetapi $a_i > a_j$. Contoh untuk film A, B, C, D, dan E:

Film	A	B	C	D	E
Ranking saya	1	2	3	4	5
Ranking X	1	3	4	2	5

Inversi: (3, 2) dan (4, 2)

Film	A	B	C	D	E
Ranking saya	1	2	3	4	5
Ranking Y	1	2	4	3	5

Inversi (4, 3)

Karena jumlah inversi dengan Y lebih sedkit daripada X, maka kesukaan saya lebih mirip dengan Y.

Anda diminta menyelesaikan persoalan inversi sbb: Diberikan sebuah senarai *A* dengan *n* elemen. Hitunglah jumlah inversi di dalam senarai tersebut. Definisi inversi: Jika $i < j$ tetapi $A[i] > A[j]$ maka pasangan $(A[i], A[j])$ disebut inversi. Contoh: A = [1, 9, 6, 4, 5], maka jumlah inversi adalah 5, yaitu pasangan (9, 6), (9, 4, (9, 5), (6, 4), dan (6, 5).

(a) Hitung jumlah inversi dan pasangan inversinya pada senarai A = [1, 5, 4, 8, 10, 2, 6, 9, 3, 7]. **(7,5)**

(b) Jika persoalan inversi diselesaikan dengan algoritma *Brute-Force*, bagaimana langkah-langkahnya? Jelaskan! Berapa jumlah perbandingan elemen yang dibutuhkan dan berapa kompleksitas algoritma dalam notasi *Big-Oh*? **(7,5)**

(c) Jika diselesaikan dengan algoritma *Divide and Conquer*, bagaimana langkah-langkahnya? Jelaskan dengan contoh senarai delapan elemen! Berapa jumlah perbandingan elemen yang dibutuhkan dan berapa kompleksitas algoritma dalam notasi *Big-Oh*? Apakah kompleksitas algoritmanya lebih baik dari *brute force*? **(10)**

(d) Jika digunakan algoritma *Mergesort* dalam menyelesaikan masalah ini, bagaimana caranya? Jelaskan dengan contoh senarai delapan elemen! Berapa kompleksitas algoritmanya? Apakah kompleksitas algoritmanya lebih baik dari jawaban b dan c? **(10)**

DFS and BFS

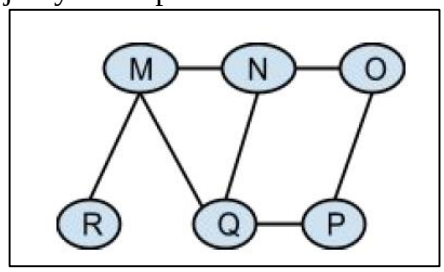
2. Terdapat dua bagian soal sebagai berikut.

(a) Dari setiap pernyataan di bawah ini, tentukan apakah pernyataan tersebut benar atau salah. Jika benar cukup tuliskan ‘Benar’, jika salah, tuliskan pernyataan yang seharusnya sehingga menjadi pernyataan yang benar.

i. Cara kerja algoritma DFS seperti struktur data *queue*, dan cara kerja algoritma BFS seperti struktur data *stack*. **(Nilai 3)**

ii. Penelusuran secara DFS pada sebuah graf berarah, akan menghasilkan pohon-pohon penelusuran yang sama jika dimulai dari sembarang simpul pada graf tersebut. **(Nilai 3)**

iii. Terdapat graf pada Gambar 1, dan penelusuran secara BFS pada graf tersebut jika dimulai dari simpul Q (dengan *expand* selanjutnya memperhatikan urutan alfabet) adalah QMNPOR. **(Nilai 3)**

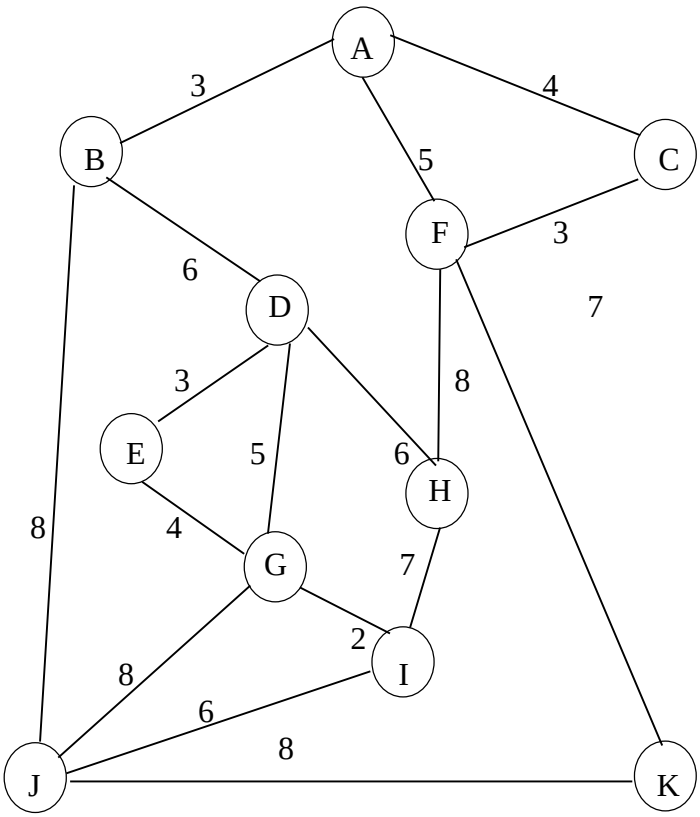


Gambar 1. Graf Tidak Berarah

(b) Terdapat sebuah graf pada Gambar 2, yang menunjukkan keterhubungan antar simpul dan jarak antar simpul.

i. Tuliskan dengan lengkap tabel yang menunjukkan pohon pencarian jalur dari simpul A ke simpul G dengan cara DFS, dengan urutan pemeriksaan tetangga sesuai urutan alfabet. Tabel berisi informasi simpul ekspan dan simpul hidup. Tuliskan jalur yang dihasilkan dan jarak dari jalur yang dihasilkan. **(Nilai 5)**

ii. Tuliskan dengan lengkap tabel yang menunjukkan pohon pencarian jalur dari simpul A ke simpul G dengan cara BFS, dengan urutan pemeriksaan tetangga sesuai urutan alfabet. Tabel berisi informasi simpul ekspan dan simpul hidup. Tuliskan jalur yang dihasilkan dan jarak dari jalur yang dihasilkan. **(Nilai 5)**



Gambar 2. Graf dengan Jarak antar Simpul

Decrease and Conquer

3. Terdapat sebuah matriks A berukuran $n \; x \; n$, yang sudah terurut menaik elemen-elemennya, sedemikian sehingga $A[i][j] < A[i][j']$ untuk $j < j'$; dan $A[i][j] < A[i'][j]$ untuk $i < i'$. Persoalan yang akan

diselesaikan adalah menentukan apakah sebuah elemen x ada pada matriks tersebut. Gunakan pendekatan Decrease and Conquer untuk menyelesaikan persoalan tersebut.

(a) Tuliskan langkah-langkah pendekatan yang anda usulkan, dan tuliskan apakah pendekatan tersebut termasuk *decrease by a contant*, *decrease by a constant factor*, atau *decrease by variable size*. Tentukan juga kompleksitas pendekatan usulan anda dalam notasi Big O. **(Nilai 10)**

(b) Terapkan pendekatan usulan anda (langkah per langkah) untuk mencari apakah $x = 29$ terdapat pada matriks berikut ini, dan hasilkan posisi ditemukannya elemen tersebut. **(Nilai 6)**

$$\begin{bmatrix} 10 & 20 & 30 & 40 \\ 15 & 25 & 35 & 45 \\ 27 & 29 & 37 & 48 \\ 32 & 33 & 39 & 50 \end{bmatrix}$$

Exhaustive Search + Greedy

4. Lakukanlah penjadwalan untuk sebuah mesin yang menerima 7 *job,* dan setiap *job* diproses selama 1 satuan waktu. Fungsi objektifnya adalah memaksimumkan profit.

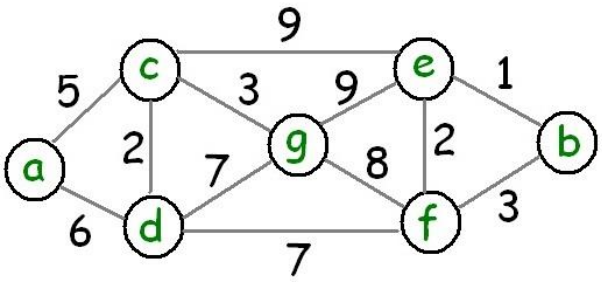
Job ke-i	1	2	3	4	5	6	7
deadline	2	4	3	2	3	1	1
profit	40	15	60	20	10	45	55

(a) Jelaskanlah proses penjadwalan dengan exhaustive search, dan tentukan kompleksitas algoritmanya? **(Nilai 7.5)**

(b) Jelaskanlah strategi greedy untuk penjadwalan ini dan berikanlah solusinya langkah per langkah. **(Nilai 7.5)**

Greedy

5. Diberikan graf berikut ini, kita akan menentukan lintasan terpendek dari simpul a ke semua simpul lainnya.



(a) Gunakanlah algoritma Kruskal untuk menentukan lintasan terpendek tersebut. Sebelum mengerjakan, tuliskanlah strategi greedy yang digunakan Kruskal. **(Nilai 7.5)**

(b) Gunakanlah algoritma Dijkstra untuk menentukan lintasan terpendek tersebut. Sebelum mengerjakan, tuliskanlah strategi greedy yang digunakan Dijkstra. **(Nilai 10)**

(c) Berikanlah kesimpulan dari hasil (a) dan (b). **(Nilai 2.5)**