

Berdoalah terlebih dahulu agar Anda sukses dalam ujian ini!

Algoritma Runut-balik (*backtracking*)

1. Terdapat persoalan *cryptarithmic* berikut ini. **(Nilai 15)**

$$\begin{array}{r}
 T W O \\
 + T W O \\
 \hline
 F O U R
 \end{array}$$

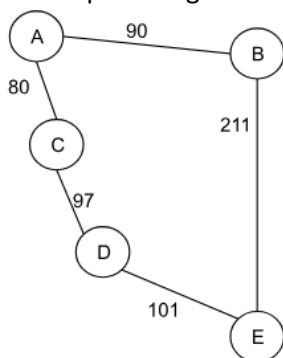
Setiap huruf harus diganti dengan sebuah angka (huruf yang sama merepresentasikan angka yang sama, huruf yang berbeda harus merepresentasikan angka yang berbeda), sedemikian sehingga ketika sudah dalam bentuk rangkaian angka, ekspresi matematika menunjukkan penjumlahan yang benar. Huruf pertama pada setiap kata tidak boleh diganti dengan angka nol.

Dengan memanfaatkan algoritma *Backtracking* untuk persoalan *cryptarithmic* tersebut, tentukan:

- a. Representasi dari solusi persoalan
- b. Fungsi Pembangkit
- c. Batasan-batasan (*constraints*) dari persoalan tersebut agar solusi yang diberikan benar.

Catatan: Jawaban untuk butir (a) hingga butir (c) tidak perlu sampai menyelesaikan (menghasilkan solusi) persoalan *cryptarithmic* tersebut."

2. Terdapat graf tidak berarah dengan jarak antar simpul sebagai berikut.



Dari graf tersebut ingin dicari jalur terpendek dari simpul A ke simpul E. Jika diperlukan, heuristik suatu simpul n atau $h(n)$ dari persoalan tersebut adalah banyaknya sisi/ busur minimal yang menghubungkan simpul n dengan simpul tujuan. Pencarian solusi menggunakan Uniform Cost Search (UCS), Greedy Best First Search, atau A*; dengan iterasi pertama untuk semua teknik pencarian adalah sebagai berikut.

Iterasi	Simpul Ekspan	Simpul Hidup
1	A	B, C

Proses pencarian dihentikan ketika simpul tujuan sudah menjadi simpul ekspan. **(Nilai 15)**

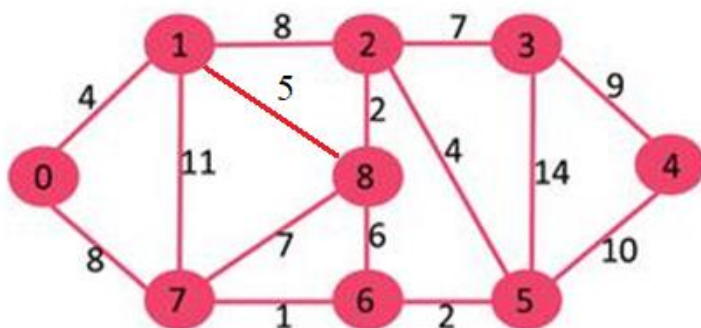
- Tentukan nilai heuristik untuk semua simpul pada graf, dan sebutkan untuk setiap simpul apakah nilai heuristiknya *admissible* atau tidak.
- Pada iterasi ke-4, sebutkan untuk setiap teknik pencarian (3 teknik di soal), simpul apa yang menjadi simpul ekspansi.
- Pada iterasi ke-4, sebutkan untuk setiap teknik pencarian (3 teknik di soal), simpul apa yang menjadi simpul hidup dan nilai $f(n)$ untuk setiap simpul hidup pada setiap teknik pencarian.
- Untuk setiap teknik pencarian, sebutkan pada iterasi ke berapa proses pencarian dihentikan.
- Untuk setiap teknik pencarian, tuliskan jalur hasil pencarian dan total jaraknya.

Algoritma Branch and Bound

- Diketahui pada sebuah pesawat ulang alik dapat membawa jet booster seberat 150 satuan. Terdapat 5 tipe jet: Jet A yang mempunyai berat 15 dan kemampuan *boost* 14, Jet C yang mempunyai berat 67 dan kemampuan *boost* 32, Jet E yang mempunyai berat 8 dan kemampuan *boost* 4, Jet G yang mempunyai berat 75 dan kemampuan *boost* 52, dan Jet K yang mempunyai berat 30 dan kemampuan *boost* 7. Tentukan jet mana saja yang dibawa yang akan memberikan *boosting* maksimum dengan algoritma *Branch and Bound*. **(Nilai 10)**

Bagian Programma dinamis

- Seekor semut berjalan dari titik 0 ke titik 4. Tentukan rute terpendek yang dilalui oleh semut itu dengan algoritma programma dinamis (*dynamic programming*) dan panjang rute. Dalam menjawab soal ini harus menggunakan tabel perhitungan pada setiap tahap dan rekonstruksi solusi pada akhir tahap. **(Nilai 15)**



Bagian Teori P, NP, NP Complete

- Tuliskan apakah soal di bawah ini **benar** atau **salah** (setiap soal bernilai 1,5):
 - Persoalan “diberikan sebuah matriks M berukuran $n \times n$, tentukan apakah M memiliki nilai eigen” termasuk ke dalam kelas *NP*.
 - Persoalan *minimum spanning tree* termasuk ke dalam kelas *P*.
 - Jika X adalah dua persoalan di dalam kelas *NP-complete*, Z adalah persoalan di dalam kelas *NP* tetapi bukan *NP-complete*, maka jika suatu hari nanti Z dapat diselesaikan dalam waktu polinom, maka X juga dapat diselesaikan dalam waktu polinom.
 - Dengan mengasumsikan $P \neq NP$, maka $NP-complete = P$
 - Persoalan SAT (*satisfiable problem*) termasuk ke dalam kelas *NP-complete*
 - Diberikan persoalan SAT dengan dua peubah, a dan b . Fungsi $C = a \wedge \bar{b}$ adalah *satisfiable*.
 - Diberikan persoalan SAT dengan dua peubah, a dan b . Fungsi $C = a \wedge \bar{a} \wedge b$ tidak *satisfiable*.
 - Algoritma non-deterministik memiliki tahap verifikasi dalam waktu non-polinomial
 - Kelas *P* adalah himpunan bagian dari *NP* karena *P* tidak memiliki tahap verifikasi

j) Persoalan “P versus NP” adalah persoalan keputusan yang waktu penyelesaiannya non-polinomial namun solusinya dapat diverifikasi dalam waktu polinomial.

Pencocokan string dan regex

6. Untuk algoritma KMP, diketahui substring DNA sbb : GCAGAGAG. (Nilai 10)

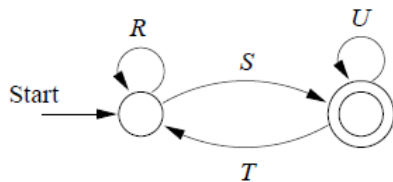
- a. Buatlah table border function nya.
- b. Jika diketahui urutan string DNA sbb : GCATCGCAGAGAGTATACAGTACG, ilustrasikan Langkah-langkah perbandingan karakter per karakter di algoritma KMP untuk mencari substring di atas pada string tersebut untuk mencari jumlah operasi perbandingan character yang dilakukan.
- c. Berapakah jumlah operasi perbandingan character nya total ?

7. Untuk algoritma Boyer Moore, diketahui substring DNA sbb : GCAGAGAG. (Nilai 10)

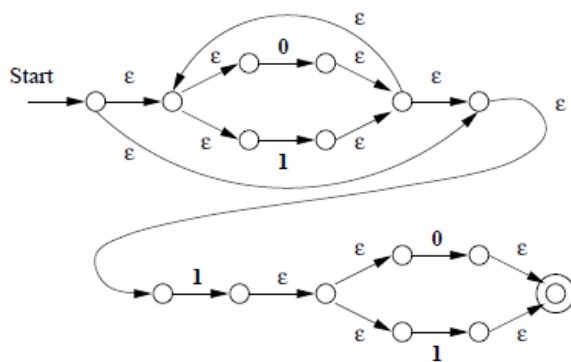
- a. Buatlah table Last-occurrence function (L()) nya.
- b. Jika diketahui urutan string DNA sbb : GCATCGCAGAGAGTATACAGTACG, ilustrasikan Langkah-langkah perbandingan karakter per karakter di algoritma Boyer-Moore untuk mencari substring di atas pada string tersebut untuk mencari jumlah operasi perbandingan character yang dilakukan.
- c. Berapakah jumlah operasi perbandingan character nya total ?

8. Ubahlah menjadi sebuah REGEX dengan notasi sesuai dengan yang di slide untuk finite automata sbb :

a. Finite automata



b. Finite automata dengan epsilon sbb :



(Nilai 10)

9. Diberikan regex (regular expression) $m[ae]n?[dsk]^*[jtiudf]$. Tentukanlah kata yang cocok dengan regex tersebut pada himpunan kata berikut:

mandi, mesti, mati, mat, mantu, messi, medi, mansu,
maki, maen, massif, mesu, mantap, masjid, mensu

(Nilai 5)

10. Tentukan prediksi nilai anda untuk kuliah ini (A/AB/B/BC/C/D/E)

(Nilai = 2)

Total nilai = 107