

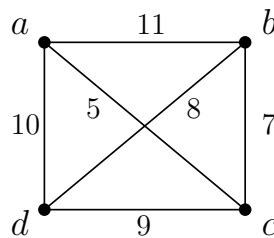
## EXERCISE 2.3: EXHAUSTIVE SEARCH (PENCARIAN MENYELURUH)

dikerjakan pada saat perkuliahan tatap muka

### 1. Traveling Salesman Problem

*Hamiltonian cycle* adalah *cycle* (sirkuit) yang mengunjungi setiap simpul pada graf tepat satu kali.

1. Buatlah sebuah algoritma brute-force untuk “menemukan sirkuit Hamilton dengan bobot minimum”.
2. Ilustrasikan algoritma Anda pada graf berikut.



3. Hitunglah kompleksitas waktu algoritma rancangan Anda.

### 2. 1/0 Knapsack Problem

1. Diberikan  $n$  objek dan ransel berkapasitas  $K$ . Setiap objek  $i$  memiliki bobot  $w_i$  dan untung  $p_i$ .
2. Buatlah sebuah algoritma brute-force untuk menyeleksi objek ke dalam ransel agar keuntungan maksimal. Berat total benda tidak boleh melebihi kapasitas ransel.
3. Apa makna “1/0” pada 1/0 Knapsack Problem?
4. Diberikan empat objek dan ransel berkapasitas  $K = 16$ . Karakteristik dari setiap objek dirangkum dalam tabel berikut:

Object	Weight	Profit
1	5	10
2	5	30
3	9	40
4	2	20

5. Hitunglah kompleksitas waktu algoritma rancangan Anda.

### 3. Assignment problem

Diberikan  $n$  staf dan  $n$  tugas. Setiap staff diberikan sebuah tugas. Staff ( $s_i$ ) ditugaskan ke tugas ( $t_j$ ) dengan biaya  $c(i, j)$ .

1. Mengapa algoritma dibutuhkan untuk menyelesaikan permasalahan ini?
2. Rancang algoritma brute force untuk menetapkan tugas sedemikian rupa sehingga total biaya  $\sum c(i, j)$  diminimalkan.
3. Diberikan 4 staf dan 4 tugas (*assignment*) dengan matriks biaya sebagai berikut. Terapkan algoritma rancangan Anda untuk menyelesaikan permasalahan berikut.

$$C = \begin{bmatrix} \text{task 1} & \text{task 2} & \text{task 3} & \text{task 4} \\ 9 & 2 & 7 & 8 \\ 6 & 4 & 3 & 7 \\ 5 & 8 & 1 & 8 \\ 7 & 6 & 9 & 4 \end{bmatrix} \begin{matrix} \text{staf a} \\ \text{staf b} \\ \text{staf c} \\ \text{staf d} \end{matrix}$$

#### 4. Partition problem

*Permasalahan partisi* didefinisikan sebagai berikut:

Diberikan sebuah himpunan yang terdiri dari  $n$  bilangan bulat positif. Partisi himpunan tersebut menjadi dua himpunan terpisah sedemikian rupa sehingga jumlah kedua subhimpunan itu sama.

**Contoh.** Diberikan  $n = 6$ , dan himpunan bilangan: 3, 8, 4, 6, 1, 2. Himpunan bilangan tersebut dapat dibagi menjadi dua subhimpunan yaitu: {3, 8, 1} dan {4, 6, 2}, dimana jumlah masing-masingnya adalah 12.

1. Coba diskusikan sebuah contoh riil yang melibatkan partition problem.
2. Buatlah sebuah algoritma brute-force untuk menyelesaikan permasalahan tersebut.
3. Terapkan algoritma rancangan Anda pada himpunan berikut:

$$S = \{1, 2, 3, 4, 5, 6\}$$

4. Tentukan kompleksitas waktu dari algoritma yang Anda rancang.