

# 1 Knapsack Problem

Masalah Knapsack adalah masalah yang terkenal dalam algoritma. Diberikan  $n$  items/objek yang memiliki properti seperti bobot  $w_1, w_2, \dots, w_n$  dan keuntungan  $p_1, p_2, \dots, p_n$ . Dengan kapasitas bobot Knapsack  $K$ . Permasalahan yang terjadi adalah bagaimana memilih memilih objek - objek yang dimasukkan ke dalam knapsack sedemikian sehingga memaksimalkan keuntungan. Total bobot objek yang dimasukkan ke dalam knapsack tidak boleh melebihi kapasitas knapsack  $K$ .

*If you do not like the idea of putting yourself in the shoes of a thief who wants to steal the most valuable loot that fits into his knapsack, think about a transport plane that has to deliver the most valuable set of items to a remote location without exceeding the plane's capacity.*

Persoalan 0/1 Knapsack dapat kita pandang sebagai mencari himpunan bagian (subset) dari keseluruhan objek yang muat ke dalam knapsack dan memberikan total keuntungan terbesar.

Solusi persoalan dinyatakan sebagai vektor  $n$ -tupel:

$$X = \{x_1, x_2, \dots, x_n\}$$

dengan  $x_i = 1$  jika objek ke- $i$  dimasukkan ke dalam knapsack,  $x_i = 0$  jika objek ke- $i$  tidak dimasukkan.

Formulasi matematisnya dapat dibentuk menjadi:

$$\text{Maximumkan } F = \sum_i^n p_i x_i$$

dengan kendala

$$\sum_i^n w_i x_i \leq K$$

dengan  $i = 0$  atau  $1, \forall i = 1, 2, \dots, n$ .

Algoritma exhaustive search untuk persoalan 0/1 Knapsack :

1. Enumerasikan (list) semua himpunan bagian dari himpunan dengan  $n$  objek.
2. Hitung (evaluasi) total keuntungan dari setiap himpunan bagian dari langkah 1.
3. Pilih himpunan bagian yang memberikan total keuntungan terbesar.

**Example 3.1** Diberikan data untuk  $n = 4$  seperti berikut ini:

$$w_1 = 2; \quad p_1 = 20$$

$$w_2 = 5; \quad p_2 = 30$$

$$w_3 = 10; \quad p_3 = 50$$

$$w_4 = 5; \quad p_4 = 10$$

Kapasitas knapsack

$$K = 16.$$

Langkah - langkah pencarian solusi 0/1 Knapsack secara exhaustive search dirangkum dalam tabel di bawah ini :

Tabel 1: Solusi menggunakan exhaustive search, bagian tebal menyatakan solusi optimal.

Himpunan Bagian	Total Bobot	Total keuntungan
{}	0	0
{1}	2	20
{2}	5	30
{3}	10	50
{4}	5	10
{1, 2}	7	50
{1, 3}	12	70
{1, 4}	7	30
<b>{2, 3}</b>	<b>15</b>	<b>80</b>
{2, 4}	10	40
{3, 4}	15	60
{1, 2, 3}	17	tidak layak
{1, 2, 4}	12	60
{1, 3, 4}	17	tidak layak
{2, 3, 4}	20	tidak layak
{1, 2, 3, 4}	22	tidak layak

Himpunan bagian objek yang memberikan keuntungan maksimum adalah {2,3} dengan total keuntungan adalah 80. Sehingga solusi:  $X = \{0,1,1,0\}$ .

Berapa banyak himpunan bagian dari sebuah himpunan dengan  $n$  elemen? Jawabnya adalah  $2^n$ . Waktu untuk menghitung total bobot objek yang dipilih =  $O(n)$ . Sehingga , Kompleksitas algoritma exhaustive search untuk persoalan 0/1 Knapsack =  $O(n \cdot 2^n)$ .

TSP dan 0/1 Knapsack , adalah contoh persoalan eksponensial. Keduanya digolongkan sebagai persoalan **NP (Non-deterministic Polynomial)**, karena tidak mungkin dapat ditemukan algoritma polinomial untuk memecahkannya.

**Exercise 3.2** Diberikan data untuk  $n = 4$  seperti berikut ini:

$$w_1 = 7; \quad p_1 = \$42$$

$$w_2 = 3; \quad p_2 = \$12$$

$$w_3 = 4; \quad p_3 = \$40$$

$$w_4 = 5; \quad p_4 = \$25$$

Kapasitas knapsack  $K = 10$ . Tentukan solusi optimalnya.

## References

1. Anany, L. (2003). Introduction to the design and analysis of algorithms. Villanova University.
2. <http://www.csd.uoc.gr/hy583/papers/ch11.pdf>
3. [Kol95] Kolman, B. and Beck, R.E. Elementary Linear Programming with Applications, 2nd ed. Academic Press, 1995.