

Penerapan Algoritma Greedy dalam Menentukan Pilihan Buah dan Sayur yang Bernutrisi Tinggi

Nando Rusrin Pratama / 13517148
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13517148@std.stei.itb.ac.id

Abstrak—Salah satu dari banyak aplikasi algoritma Greedy adalah membantu dalam menentukan pilihan buah dan sayur yang bernutrisi tinggi. Dalam menentukan pilihan, kandungan vitamin dan mineral menjadi faktor yang penting. Namun, dalam menentukan pilihan tersebut dalam kehidupan sehari-hari kita dibatasi oleh biaya juga. Karena itu, diperlukan solusi untuk menentukan pilihan buah dan sayur yang bernutrisi tinggi (optimal) dengan biaya tertentu. Masalah ini dapat diselesaikan dengan mudah dengan menggunakan metode Knapsack.

Keywords — Greedy, Knapsack, Integer Knapsack, Fractional Knapsack, Buah, Sayur, Nutrisi, Vitamin, Mineral

I. PENDAHULUAN

Vitamin dan mineral sangat diperlukan dalam menjaga kesehatan tubuh. Keduanya berperan penting dalam proses pertumbuhan dan metabolisme tubuh. Vitamin dan mineral berperan dalam membantu pembentukan energi. Energi berasal dari karbohidrat, lemak, dan protein. Akan tetapi, tanpa vitamin dan mineral, ketiganya sulit diubah menjadi energi.

Konsumsi vitamin dan mineral juga berfungsi dalam menangkal radikal bebas di tubuh yang bisa menjadi sumber penyakit. Vitamin dan mineral juga membantu pelepasan neurotransmitter. Neurotransmitter diperlukan untuk mengirim perintah dari otak. Jika kekurangan vitamin dan mineral, maka perintah dari otak akan sulit dilakukan.

Ada dua jenis vitamin, yaitu yang larut dalam air dan larut lemak. Vitamin yang larut lemak adalah A, D, E, dan K. Vitamin lainnya, seperti B dan C adalah vitamin larut air. Sementara itu, mineral antara lain kalsium, asam folat, fosfor, iodine, magnesium, dan zinc.

Untuk memenuhi kebutuhan vitamin dan mineral yang dibutuhkan tubuh, kita dapat mengonsumsi buah-buahan dan sayur-sayuran. Vitamin dan mineral banyak terdapat pada buah dan sayuran. Buah maupun sayuran ini pun seharusnya dikonsumsi mulai dari makan pagi, siang, dan malam setiap hari. Sayangnya, tak semua orang memiliki kebiasaan konsumsi buah dan sayuran.

Dalam pemenuhan kebutuhan vitamin dan mineral tentu ada biaya yang dikeluarkan. Selanjutnya, akan dijelaskan cara menentukan pilihan buah dan sayuran bernutrisi tinggi untuk vitamin A, C, dan asam folat (B9) dengan biaya tertentu

menggunakan salah satu metode algoritma Greedy yaitu metode Knapsack.

II. DASAR TEORI

A. Algoritma Greedy

Algoritma Greedy merupakan salah satu algoritma yang dapat digunakan dalam memecahkan persoalan optimasi. Persoalan optimasi dapat dibedakan menjadi 2 yaitu maksimasi dan minimasi. Maksimasi adalah mendapatkan solusi yang hasilnya semaksimal mungkin dari suatu persoalan. Sedangkan, minimasi adalah mendapatkan solusi yang hasilnya seminimal mungkin dari suatu persoalan. Algoritma greedy umumnya mencari solusi optimum lokal disetiap langkah dengan harapan mendapat solusi optimum global diakhir langkah.

Umumnya contoh persoalan maksimasi adalah untuk memperoleh keuntungan sebesar mungkin dari suatu persoalan. Sedangkan, untuk contoh persoalan minimasi umumnya adalah masalah waktu, yaitu bagaimana cara mencapai tujuan dengan waktu paling sedikit atau menghemat waktu.

Prinsip greedy adalah “take what you can get now”, artinya disetiap langkah algoritma ini akan mengambil langkah yang paling menguntungkan pada tahap itu tanpa peduli langkah selanjutnya. Solusi-solusi tersebut adalah solusi optimum lokal dengan harapan diakhir langkah akan mendapatkan solusi optimum global. Adapun contoh pseudo-code algoritma greedy adalah sebagai berikut :

Untuk persoalan minimasi

```
function PenjadwalanPelanggan(input C :  
himpunan_pelanggan) -> himpunan_pelanggan  
{  
    mengembalikan urutan jadwal pelayanan  
    pelanggan yang meminimumkan waktu di dalam  
    sistem  
}
```

Deklarasi

S : himpunan_pelanggan

i : pelanggan

Algoritma

S <- {}

```

while (C != {}) do
  i <- pelanggan yang mempunyai t[i]
  terkecil
  C <- C - {i}
  S <- S U {i}
endwhile
return S

```

Untuk persoalan maksimasi

```

function Greedy-Activity-Selector(input s, f)
{ Asumsi: aktivitas sudah diurut terlebih
dahulu
Berdasarkan waktu selesai: f1 <= f2 <= ... <= fn
}

```

Algoritma

```

n ← length(s)
A ← {1}
j ← 1
for i ← 2 to n do
  if si >= fj then
    A ← A U {i}
    j ← i
  end
end

```

Penjelasan algoritma greedy :

1. Ambil salah satu kandidat yang memenuhi kategori greedy, lalu masukan kedalam himpunan solusi.
2. Setelah himpunan kandidat diambil dan dimasukkan kedalam solusi, kurangi jumlah himpunan.
3. Jika layak, masukkan ke dalam himpunan solusi.
4. Jika tidak layak jangan masukkan kedalam himpunan solusi. (definisi layak & tidak layak didefinisikan oleh fungsi kelayakan)
5. Lakukan hingga sudah tidak ada yang tersisa dari himpunan kandidat

B. Elemen-Element Algoritma Greedy

Algoritma Greedy dalam menyelesaikan suatu persoalan, terdiri dari 5 buah elemen, antara lain :

1. Himpunan Kandidat, C.

Berisi semua entitas yang membentuk solusi. Pada setiap langkah sebuah entitas akan dipilih dari himpunan kandidat.

2. Himpunan Solusi, S.

Himpunan solusi adalah entitas-entitas yang menjadi solusi permasalahan. Dalam proses membentuk himpunan solusi, pada setiap langkah kita membutuhkan fungsi seleksi dan fungsi kelayakan.

3. Fungsi Seleksi

Fungsi seleksi adalah fungsi yang menjelaskan bagaimana cara kita memilih solusi optimum lengkap.

4. Fungsi Kelayakan

Fungsi kelayakan dapat dinyatakan dengan predikat lengkap. Fungsi kelayakan adalah fungsi syarat apakah solusi yang kita ambil layak untuk dimasukkan ke dalam himpunan solusi. Artinya solusi tersebut tidak melanggar constraint yang ada.

5. Fungsi Obyektif

Fungsi obyektif adalah fungsi yang dapat menjelaskan kondisi solusi sudah optimum atau tidak.

C. Metode Knapsack

Knapsack adalah persoalan optimasi yang sering terjadi dalam kehidupan sehari-hari. Persoalan yang sering menggunakan knapsack adalah delivery atau antar barang. Supaya biaya untuk mengantar barang dapat diminimalisir, maka jumlah dan berat barang yang dibawa harus dimaksimalkan. Hal tersebut bertujuan agar tidak perlu berkali-kali bolak balik untuk mengantar barang ke tempat tujuan.

Algoritma greedy dapat digunakan untuk menyelesaikan persoalan integer knapsack tersebut. Selain untuk mengantar barang, persoalan knapsack ini juga dapat diterapkan pada pencurian barang. Pada saat mencuri ada beberapa batasan yang perlu diperhatikan, paling tidak antara lain jumlah barang yang bisa dibawa dan berat total barang yang dapat dibawa.



Gambar 2.1 Ilustrasi Persoalan Knapsack

Penyelesaian dengan algoritma greedy, memasukkan objek satu per satu ke dalam knapsack. Sekali objek dimasukkan ke dalam knapsack, objek tersebut tidak bisa dikeluarkan lagi.

Berikut contoh pseudo algoritma heuristik Integer Knapsack.

$$\text{Maksimasi } F = \sum_{i=1}^n p_i x_i$$

dengan kendala (*constraint*)

$$\sum_{i=1}^n w_i x_i \leq K$$

yang dalam hal ini,
 $x_i = 0$ atau 1 , $i = 1, 2, \dots, n$

Terdapat beberapa strategi greedy yang heuristik yang dapat digunakan untuk memilih objek yang akan dimasukkan ke dalam knapsack:

1. Greedy by profit.
 - Pada setiap langkah, pilih objek yang mempunyai keuntungan terbesar.
 - Mencoba memaksimalkan keuntungan dengan memilih objek yang paling menguntungkan terlebih dahulu.
2. Greedy by weight.
 - Pada setiap langkah, pilih objek yang mempunyai berat teringan.
 - Mencoba memaksimalkan keuntungan dengan memasukkan sebanyak mungkin objek ke dalam knapsack.
3. Greedy by density.
 - Pada setiap langkah, knapsack diisi dengan objek yang mempunyai p_i/w_i terbesar.
 - Mencoba memaksimalkan keuntungan dengan memilih objek yang mempunyai keuntungan per unit berat terbesar.

Pemilihan objek berdasarkan salah satu dari ketiga strategi di atas tidak menjamin akan memberikan solusi optimal. Untuk itu, ada algoritma Fractional Knapsack yang menggunakan pemilihan objek berdasarkan p_i/w_i terbesar yang pasti menghasilkan solusi yang optimum. Berikut algoritma persoalan fractionalknapsack:

1. Hitung harga p_i/w_i , $i = 1, 2, \dots, n$
2. Urutkan seluruh objek berdasarkan nilai p_i/w_i dari besar ke kecil
3. Panggil FractionalKnapsack

function FractionalKnapsack(input C : himpunan_objek, K : real) -> himpunan_solusi

{ Menghasilkan solusi persoalan fractional knapsack dengan algoritma greedy yang menggunakan strategi pemilihan objek berdasarkan density (p_i/w_i). Solusi dinyatakan sebagai vektor $X = x[1], x[2], \dots, x[n]$.

Asumsi: Seluruh objek sudah terurut berdasarkan nilai p_i/w_i yang menurun }

Deklarasi

i , TotalBobot : integer
 MasihMuatUtuh : boolean
 x : himpunan_solusi

Algoritma:

```

for i <- 1 to n do
  x[i] <- 0 { inisialisasi setiap fraksi objek i dengan 0 }
endfor

i <- 0
TotalBobot <- 0
MasihMuatUtuh <- true
while (i <= n) and (MasihMuatUtuh) do {
  tinjau objek ke-i }
  i <- i + 1
  if TotalBobot + C.w[i] <= K then
    { masukkan objek i ke dalam knapsack }
    x[i] <- 1
    TotalBobot <- TotalBobot + C.w[i]
  else
    MasihMuatUtuh <- false
    x[i] <- (K - TotalBobot)/C.w[i]
  endif
endwhile
{ i > n or not MasihMuatUtuh }

return x
  
```

III. ANALISIS PERSOALAN

Vitamin dan mineral sangat diperlukan dalam menjaga kesehatan tubuh. Keduanya berperan penting dalam proses pertumbuhan dan metabolisme tubuh. Untuk memenuhi kebutuhan vitamin dan mineral yang dibutuhkan tubuh, kita dapat mengonsumsi buah-buahan dan sayur-sayuran.

Menurut https://www.health.harvard.edu/staying-healthy/listing_of_vitamins manusia membutuhkan Vitamin A (0.9 mg/hari), Vitamin C (90 mg/hari), dan Asam Folat (0.2 mg/hari). Karena vitamin A dan C serta asam folat adalah vitamin dan mineral yang sering ditemukan dalam buah dan sayur, maka digunakan data kebutuhan harian untuk tiga vitamin di atas saja.

Jika setiap hari, seseorang diharuskan makan buah dan sayur sebelum makan pagi, makan siang, dan makan malam agar nutrisinya tercukupi, maka sebelum sarapan, makan siang dan makan malam, orang itu harus mengonsumsi buah dan sayur dengan kandungan total Vitamin A (3 x 0.3 mg), Vitamin C (3 x 30 mg) dan asam folat (3 x 0.067 mg).

Dalam mengonsumsi vitamin dan mineral juga ada batasan tertentu. Semakin kita mendekati batas itu, maka akan semakin beresiko mengalami efek samping. Contohnya, konsumsi vitamin A berlebihan dapat menyebabkan masalah pada tulang. Lalu, konsumsi berlebihan vitamin C dapat menyebabkan masalah pencernaan. Dan konsumsi asam folat

berlebihan dapat menyebabkan kerusakan saraf. Untuk itu, perlu diperhatikan jumlah konsumsi buah dan sayur harian.

Namun, kekurangan vitamin dan mineral juga dapat menyebabkan gangguan-gangguan kesehatan. Untuk itu, diperlukan sebuah algoritma yang mengoptimalkan konsumsi buah dan sayur harian yang bernutrisi tinggi dengan biaya tertentu.

Melalui survey harga dan data dari [https:// www.lenntech.com/fruit-vegetable-vitamin-content.htm](https://www.lenntech.com/fruit-vegetable-vitamin-content.htm), berikut daftar harga buah dan sayur yang sering ditemui beserta vitamin yang terkandung di dalamnya.

Nama Buah dan Sayur	Vitamin A (mg)	Vitamin C (mg)	Asam Folat (mg)	Harga (Rp)
Apel	0.005	5	0.003	3000
Pisang	0.008	10	0.016	3000
Anggur	0.028	40	0.015	5000
Kiwi	0.007	70	0.023	7000
Lemon	0.001	40	0.008	3000
Mangga	0.053	23	0.023	3000
Jeruk	0.012	49	0.018	2000
Nanas	0	25	0.004	4000
Tomat	0.088	15	0.008	500
Semangka	0.045	6	0.001	3000
Alpukat	0.11	17	0.008	5000
Brokoli	0.2	110	0.012	3000
Kubis	0.01	80	0.069	1500
Wortel	0.6	1	0.016	2000
Kembang Kol	0.001	80	0.044	3000
Jagung	0.007	0	0.034	3000
Paprika	0.022	70	0.055	2000
Bawang Perai	0.094	20	0.011	2000
Jamur	0	5	0.032	4000
Kacang Polong	0.3	1	0.022	2000
Kentang	0	14	0.023	2000
Lobak	0.05	20	0.028	2000
Bayam	0.24	25	0.1	1000

Tabel 3.1 Kandungan Vitamin dan Harga

IV. PEMODELAN MASALAH

A. Menentukan Himpunan Kandidat

Himpunan kandidat merupakan kumpulan semua entitas yang dapat menjadi solusi suatu permasalahan. Dalam permasalahan ini, kita dapat mengambil referensi dari survey harga dan data dari [https:// www.lenntech.com/fruit-vegetable-vitamin-content.htm](https://www.lenntech.com/fruit-vegetable-vitamin-content.htm). Agar tidak terlalu banyak himpunan kandidat yang diseleksi, maka data buah dan sayur adalah yang sering ditemui dalam kehidupan sehari-hari.

B. Menentukan Fungsi Seleksi

Fungsi seleksi merupakan fungsi yang melakukan pengecekan syarat kepada himpunan kandidat yang dapat masuk ke dalam himpunan solusi. Dalam menyelesaikan persoalan ini, fungsi seleksi yang tepat dapat dibuat sesuai dengan keinginan kita. Misalnya untuk persoalan ini penulis menggunakan seleksi dengan value total nutrisi per 1000 rupiah. Dengan perhitungan value sebagai berikut :

$$\text{value nutrisi per Rp 1000} = ((\text{mg vit.A} / 3) + (\text{mg vit.C} / 2000) + (\text{mg as. folat} / 1)) / (\text{harga} / 1000)$$

Nama Buah dan Sayur	Vitamin A (mg)	Vitamin C (mg)	Asam Folat (mg)	Harga (Rp)	Value Nutrisi per Rp 1000
Apel	0.005	5	0.003	3000	0.002389
Pisang	0.008	10	0.016	3000	0.007889
Anggur	0.028	40	0.015	5000	0.008867
Kiwi	0.007	70	0.023	7000	0.008619
Lemon	0.001	40	0.008	3000	0.009444
Mangga	0.053	23	0.023	3000	0.017389
Jeruk	0.012	49	0.018	2000	0.02325
Nanas	0	25	0.004	4000	0.004125
Tomat	0.088	15	0.008	500	0.089667
Semangka	0.045	6	0.001	3000	0.006333
Alpukat	0.11	17	0.008	5000	0.010633
Brokoli	0.2	110	0.012	3000	0.044556
Kubis	0.01	80	0.069	1500	0.074889
Wortel	0.6	1	0.016	2000	0.10825
Kembang Kol	0.001	80	0.044	3000	0.028111
Jagung	0.007	0	0.034	3000	0.012111
Paprika	0.022	70	0.055	2000	0.048667

Nama Buah dan Sayur	Vitamin A (mg)	Vitamin C (mg)	Asam Folat (mg)	Harga (Rp)	Value Nutrisi per Rp 1000
Bawang Perai	0.094	20	0.011	2000	0.026167
Jamur	0	5	0.032	4000	0.008625
Kacang Polong	0.3	1	0.022	2000	0.06125
Kentang	0	14	0.023	2000	0.015
Lobak	0.05	20	0.028	2000	0.027333
Bayam	0.24	25	0.1	1000	0.1925

Tabel 4.1 Kandungan Vitamin, Harga, dan Value Nutrisi

C. Menentukan Fungsi Kelayakan

Fungsi kelayakan adalah fungsi yang menentukan apakah sebuah entitas layak untuk masuk dalam himpunan solusi atau tidak. Untuk persoalan ini, dapat kita misalkan dalam membahas tentang solusi greedy apa saja yang boleh masuk dalam himpunan solusi. Misalnya untuk persoalan ini penulis menggunakan kelayakan yaitu buah dan sayur yang memiliki value nutrisi per Rp 1000 termasuk tertinggi daripada lainnya lalu dimasukkan kedalam knapsack berbatas Rp 5000.

V. PENERAPAN ALGORITMA GREEDY

Setelah selesai menentukan elemen-elemen algoritma greedy yang kita butuhkan, kita dapat lakukan langkah terakhir dengan menentukan jenis greedy yang ingin digunakan. Berikut adalah tabel value yang telah diurutkan.

Nama Buah dan Sayur	Vitamin A (mg)	Vitamin C (mg)	Asam Folat (mg)	Harga (Rp)	Value Nutrisi per Rp 1000
Bayam	0.24	25	0.1	1000	0.1925
Wortel	0.6	1	0.016	2000	0.10825
Tomat	0.088	15	0.008	500	0.089667
Kubis	0.01	80	0.069	1500	0.074889
Kacang Polong	0.3	1	0.022	2000	0.06125
Paprika	0.022	70	0.055	2000	0.048667
Brokoli	0.2	110	0.012	3000	0.044556
Kembang Kol	0.001	80	0.044	3000	0.028111

Nama Buah dan Sayur	Vitamin A (mg)	Vitamin C (mg)	Asam Folat (mg)	Harga (Rp)	Value Nutrisi per Rp 1000
Lobak	0.05	20	0.028	2000	0.027333
Bawang Perai	0.094	20	0.011	2000	0.026167
Jeruk	0.012	49	0.018	2000	0.02325
Mangga	0.053	23	0.023	3000	0.017389
Kentang	0	14	0.023	2000	0.015
Jagung	0.007	0	0.034	3000	0.012111
Alpukat	0.11	17	0.008	5000	0.010633
Lemon	0.001	40	0.008	3000	0.009444
Anggur	0.028	40	0.015	5000	0.008867
Jamur	0	5	0.032	4000	0.008625
Kiwi	0.007	70	0.023	7000	0.008619
Pisang	0.008	10	0.016	3000	0.007889
Semangka	0.045	6	0.001	3000	0.006333
Nanas	0	25	0.004	4000	0.004125
Apel	0.005	5	0.003	3000	0.002389
Lobak	0.05	20	0.028	2000	0.027333
Bawang Perai	0.094	20	0.011	2000	0.026167
Jeruk	0.012	49	0.018	2000	0.02325
Mangga	0.053	23	0.023	3000	0.017389
Kentang	0	14	0.023	2000	0.015
Jagung	0.007	0	0.034	3000	0.012111

Tabel 5.1 Kandungan Vitamin, Harga, dan Value Nutrisi Terurut

Lalu setelah diurutkan, kita masukkan satu-persatu buah dan sayur ke dalam knapsack. Jika makanan yang dimasukkan akan melebihi batas harga knapsack, ganti dengan bawahnya sehingga menghasilkan buah dan sayur sebagai berikut :

	Vit. A	Vit. C	Asam Folat	Harga
Bayam	0.24	25	0.1	1000
Wortel	0.6	1	0.016	2000
Tomat	0.088	15	0.008	500
Kubis	0.01	80	0.069	1500
Total	0.938	121	0.193	5000

Dengan total 0.938 mg vitamin A, 121 mg vitamin C, dan 0.193 mg asam folat. Dapat dikatakan jika nutrisi harian untuk vitamin A (0.9 mg), vitamin C (90 mg), dan asam folat (0.2 mg) sudah terpenuhi. Penerapan greedy pun dapat dikatakan cukup efektif.

KESIMPULAN

Algoritma greedy merupakan salah satu algoritma yang dapat memberikan hasil optimal lokal di setiap langkahnya dengan harapan hasil akhir juga optimal. Penerapan algoritma greedy dalam persoalan knapsack untuk menentukan pilihan buah dan sayur yang bergizi tinggi dengan biaya yang minimal bisa memberikan hasil yang optimal. Perbedaan total nutrisi antara model dan rekomendasi tidak berbeda jauh dan masih dalam batas toleransi. Batas atas konsumsi perhari untuk vitamin A (3 mg), vitamin C (2000 mg) dan asam folat (1 mg). Model yang dibuat pun dapat menangkap kendala-kendala yang realitis terkait masalah seperti pola hidup sehat dan standar global di bidang nutrisi dan kesehatan. Lebih lanjut, model ini juga membuktikan bahwa pilihan buah dan sayur dihasilkan secara dapat meningkatkan nutrisi secara keseluruhan.

REFERENSI

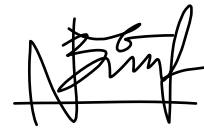
- [1] J. Van Den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, L. De Boeck, *Personnel scheduling: A literature review*. Eur. J. Oper. Res., 226 (2013), pp. 367-385.
- [2] Black, Paul E. (2 February 2005). "greedy algorithm". *Dictionary of Algorithms and Data Structures*. U.S. National Institute of Standards and Technology (NIST).

- [3] Introduction to Algorithms (Cormen, Leiserson, Rivest, and Stein) 2001, Chapter 16 "Greedy Algorithms".
- [4] Garey, Michael R.; David S. Johnson (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman. ISBN 978-0-7167-1045-5. A6: MP9, pg.247.
- [5] Kellerer, Hans; Pferschy, Ulrich; Pisinger, David (2004). *Knapsack Problems*. Springer. ISBN 978-3-540-40286-2. MR 2161720
- [6] Martello, Silvano; Toth, Paolo (1990). *Knapsack problems: Algorithms and computer implementations*. Wiley-Interscience. ISBN 978-0-471-92420-3. MR 1086874
- [7] Combs, G.F., Jr (1992). *The Vitamins: Fundamental Aspects in Nutrition and Health*. Academic Press. ISBN 978-0-12-381980-2.
- [8] Shils, M.S.; et al., eds. (2005). *Modern Nutrition in Health and Disease*. Lippincott Williams and Wilkins. ISBN 978-0-7817-4133-0.
- [9] Combs, Gerald (2008). *The vitamins: fundamental aspects in nutrition and health*. ISBN 9780121834937.
- [10] Munir, Rinaldi, Diktat Kuliah IF2211, Strategi Algoritma, Edisi Keempat, Program Studi Teknik Informatika, STEI, ITB, 2006

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 24 April 2019



Nando Rusrin Pratama – 13517148