

Penerapan Algoritma Greedy dalam Penentuan Barang Kulak di Toko Kelontong

Haidar Ihzaulhaq - 13520150

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): ihzaulhaqidar@gmail.com

Abstract— A convenience store is a place that sell various kind of daily needs. At convenience store, there are various activities, which one of these activities is restock activity. Restock activity is activity that buying goods in large quantities from distributor and resale it at convenience store. In restock activity, a convenience store owner needs to consider what items to buy and how many the quantities. It is because when they are resale the items, these items can make a maximum profit for the owner. This problem can be solved by any way, one of these solutions is by using greedy strategy. Greedy strategy is a strategy that is commonly used in optimization problems. With this strategy, it is expected that convenience store owners can get maximum profit when doing restock activities.

Keywords— *convenience store; greedy strategy; maximum; restock activities; profit;*

I. PENDAHULUAN

Toko kelontong adalah salah satu toko yang banyak tersedia di berbagai daerah di Indonesia. Menurut KBBI, kelontong berarti barang-barang untuk keperluan sehari-hari seperti sabun, sikat gigi, gelas, cangkir, dan mangkuk. Itu artinya toko kelontong adalah tempat yang menjual berbagai barang kebutuhan sehari-hari. Produk-produk yang dijual pun juga beragam dan bisa dibilang lengkap. Toko kelontong di Indonesia sendiri terbagi menjadi dua jenis, yaitu toko kelontong tradisional dan toko kelontong modern. Toko kelontong tradisional biasanya dimiliki oleh perseorangan dengan sistem jual beli yang masih memungkinkan adanya tawar-menawar dan melakukan hutang kepada penjual barang. Sedangkan, toko kelontong modern biasanya dimiliki oleh suatu lembaga berbadan hukum dan memiliki sistem jual beli yang hampir sama dengan toko swalayan, yaitu pembeli dibebaskan untuk memilih barangnya sendiri dan kemudian membayarnya dikasir. Di toko kelontong modern pun juga tidak diperbolehkan adanya kegiatan tawar-menawar dan hutang kepada penjual. Contoh dari toko kelontong modern adalah Alfamart, Indomaret, Circle-K, 7Eleven, dan lain sebagainya.

Selain ada kegiatan jual beli antara pedagang dan penjual, di toko kelontong ada juga kegiatan kulak. Kulak adalah kegiatan membeli barang dalam jumlah besar untuk dijual kembali. Kegiatan kulak ini merupakan kegiatan yang sudah pasti dilakukan oleh semua pemilik toko kelontong, baik

tradisional maupun modern. Hanya saja, ada beberapa perbedaan kegiatan kulak yang dilakukan oleh pemilik toko tradisional dan modern. Di toko kelontong modern, kegiatan kulak biasanya justru dilakukan oleh distributor barangnya. Distributor barang akan mengirimkan permintaan barang sesuai yang diminta oleh toko modern. Kegiatan kulak yang dilakukan oleh toko modern juga biasanya sudah terstruktur dan tertata dengan baik. Berbeda dengan toko modern, di toko kelontong tradisional biasanya kulak dilakukan oleh pemilik langsung ke distributor. Di sana, biasanya pemilik toko akan memilih barang yang dirasa diperlukan untuk dijual kembali di toko mereka.

Kegiatan kulak ini tentunya tidak dapat dilakukan dengan sembarangan. Kesalahan dalam memilih barang yang akan dikulak tentu akan dapat memengaruhi keuntungan yang diperoleh juga. Selain itu, jumlah barang yang dibeli juga dapat berakibat pada keuntungan yang mungkin diperoleh oleh pemilik toko. Oleh karena itu, diperlukan suatu strategi agar kegiatan kulak yang dilakukan oleh pemilik toko kelontong dapat menghasilkan keuntungan yang maksimal nantinya ketika dijual lagi di toko mereka. Salah satu strategi yang dapat diterapkan adalah strategi dengan algoritma *greedy*. Dengan algoritma *greedy* ini diharapkan pemilihan barang dan jumlah yang ditentukan oleh pemilik toko kelontong dapat memaksimalkan keuntungan dalam penjualan barang di toko mereka.

Strategi dengan algoritma *greedy* ini akan didasarkan pada data penjualan pada toko kelontong selama sebulan sebelumnya. Data-data yang diperlukan yaitu daftar nama barang yang ada di toko, harga beli barang di distributor, profit yang didapat tiap barang yang terjual, stok awal barang yang ada di toko, jumlah barang yang terjual, serta persentase barang terjual dibanding stok awal barang. Dari data-data tersebut, dapat dilakukan analisis mengenai barang apa saja yang perlu dibeli di distributor dan berapa jumlah pastinya.

Namun tentunya, perhitungan ini mungkin akan menghasilkan opsi yang tidak optimal karena dalam kegiatan jual beli memiliki banyak faktor yang dapat memengaruhinya, seperti perubahan harga jual dan perubahan profit yang mungkin terjadi di tengah-tengah waktu penjualan, kemungkinan stok barang yang rusak atau tidak layak jual, atau perubahan jenis kebutuhan yang diperlukan oleh masyarakat.

II. LANDASAN TEORI

A. Profit Penjualan

Tujuan utama dari perdagangan adalah untuk memperoleh profit penjualan, termasuk juga toko kelontong. Profit sendiri merupakan kata lain dari keuntungan. Jadi, dalam perdagangan, profit penjualan menjadi tujuan utama yang harus dicapai. Dengan diperolehnya profit penjualan, maka suatu toko akan dapat terus melakukan kegiatan perdagangan dan bahkan dapat berpeluang untuk memperbesar skala dagang yang dilakukan. Hal ini menunjukkan bahwa profit penjualan sangat penting dalam kegiatan perdagangan.

B. Algoritma Greedy

Algoritma *greedy* merupakan salah satu teknik dalam strategi penyelesaian masalah terutama yang berkaitan dengan hal optimasi permasalahan, yaitu untuk mencari nilai maksimum dari suatu permasalahan atau mencari nilai minimumnya. Algoritma *greedy* tidak termasuk dalam algoritma khusus, maksudnya algoritma *greedy* tidak memiliki ketentuan khusus dalam penerapannya. Hal ini karena algoritma *greedy* sendiri memiliki arti 'rakus' dalam bahasa Indonesia. Maksud dari 'rakus' disini adalah apapun yang dapat menguntungkan saat ini, maka ambillah hal itu sehingga pada akhirnya algoritma yang akan digunakan juga bergantung dengan kondisi permasalahan yang ingin dicapai.

Dalam algoritma *greedy*, terdapat prinsip yang selalu ditekankan, yaitu "Take what you can get now!" yang memiliki maksud ambillah sesuatu yang bisa mengoptimalkan permasalahan tanpa perlu memerdulikan konsekuensi yang mungkin akan terjadi kedepannya. Dengan konsep ini, algoritma *greedy* akan membentuk solusi yang terdiri dari langkah per langkah (*step by step*). Di setiap langkah yang diambil, akan ada banyak pilihan yang perlu dievaluasi hingga akhirnya dibuat suatu keputusan terbaik pada langkah tersebut. Keputusan yang telah diambil ini nantinya tidak dapat diubah lagi ketika sudah berlanjut pada langkah berikutnya. Jadi, pada setiap langkah yang dibuat, akan dihasilkan sebuah solusi optimum lokal dengan harapan bahwa sisa langkah yang diambil nantinya dapat mengarahkannya kepada solusi optimum global, atau solusi dari permasalahan yang dihadapi.

Suatu persoalan dapat diselesaikan dengan algoritma *greedy* apabila persoalan tersebut memiliki sifat berikut:

- Solusi optimal dari persoalan dapat diperoleh dari solusi optimal subpersoalan tersebut (langkah per langkah yang dilakukan).
- Pada setiap subpersoalan, ada suatu langkah yang bisa dilakukan yang dapat menghasilkan solusi optimal pada subpersoalan tersebut (optimum lokal). Langkah ini disebut juga dengan *greedy choice*. (Alham Fikri & William Gozali, 2019)

Dalam penggunaan algoritma *greedy*, ada beberapa elemen yang harus dipenuhi, yaitu:

1. Himpunan kandidat (C), himpunan yang berisi kandidat yang akan dipilih pada setiap langkah (misal: simpul/sisi di dalam graf, job, task, koin, karakter, dll).

2. Himpunan solusi (S), himpunan yang berisi kandidat yang sudah dipilih.
3. Fungsi solusi, adalah fungsi yang digunakan untuk menentukan apakah himpunan kandidat yang dipilih sudah memberikan solusi.
4. Fungsi seleksi (*selection function*), adalah fungsi yang digunakan untuk memilih kandidat berdasarkan strategi *greedy* tertentu yang bersifat heuristik.
5. Fungsi kelayakan (*feasible*), adalah fungsi yang digunakan untuk memeriksa apakah kandidat yang dipilih dapat dimasukkan ke dalam himpunan solusi (layak atau tidak).
6. Fungsi objektif, adalah fungsi yang digunakan untuk memaksimumkan atau meminimumkan nilai solusi.

Dengan elemen-elemen yang menyusunnya, dapat dikatakan algoritma *greedy* adalah algoritma yang melibatkan pencarian sebuah himpunan bagian S dari himpunan kandidat C dengan S harus memenuhi beberapa kriteria yang ditentukan, yaitu S menyatakan suatu solusi dan S dioptimasi oleh fungsi objektif.

Dalam *pseudocode*, algoritma *greedy* dapat dituliskan sebagai berikut:

```
function greedy (C : himpunan_kandidat) → himpunan_solusi
{fungsi dengan input himpunan kandidat dan mengembalikan himpunan solusi dari persoalan dengan algoritma greedy}

Deklarasi
x : kandidat
S : himpunan_solusi

Algoritma
S ← {} {inisialisasi dengan himpunan kosong}
while (not SOLUSI(S)) and (C ≠ {}) do
    x ← SELEKSI(C) {pilih sebuah kandidat dari C}
    C ← C - {x} {buang x dari C karena sudah dipilih}
    if LAYAK(S ∪ {x}) then {x layak untuk masuk himpunan solusi}
        S ← S ∪ {x} {tambahkan x ke himpunan solusi}
    endif
endwhile
if SOLUSI(S) then {solusi ada}
    return S
else
    write('tidak ada solusi ditemukan')
endif
```

Sebagai gambaran, ada salah satu persoalan yang dapat diselesaikan dengan algoritma *greedy* ini. Selain itu, persoalan ini juga dirasa hampir sama dengan permasalahan akan yang akan dibahas pada makalah ini. Persoalan itu bernama *integer knapsack problem*.

Integer knapsack problem adalah sebuah persoalan dengan diberikan n buah objek dan sebuah kantong (*knapsack*) dengan kapasitas bobot maksimal K . Setiap objek yang tersedia memiliki bobot sebesar w_i dan keuntungan sebesar p_i . Tujuan dari persoalan ini adalah bagaimana cara memilih objek-objek yang dimasukkan ke dalam kantong sehingga total keuntungan yang diperoleh nantinya akan maksimal dengan total bobot yang tidak melebihi kapasitas kantong. Persoalan ini diberi nama *integer* karena setiap objek yang diambil tidak dapat dipartisi atau hanya diambil sebagian, sehingga opsi dari permasalahan ini hanyalah mengambil objeknya atau tidak sama sekali.

Persoalan ini secara matematis memiliki formulasi sebagai berikut:

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

dengan kendala (*constraint*)

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

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

Gambar 2.1 Formulasi Maksimasi

Sumber : [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf)

Dengan formulasi tersebut, maka terdapat beberapa strategi *greedy* yang dapat digunakan untuk memilih objek yang akan dimasukkan ke dalam kantong:

1. *Greedy by profit*

Strategi *greedy by profit* ini dilakukan dengan memilih objek yang mempunyai keuntungan terbesar pada setiap langkahnya. Strategi ini bertujuan untuk mencoba memaksimalkan keuntungan dengan memilih objek yang menguntungkan terlebih dahulu.

2. *Greedy by weight*

Strategi *greedy by weight* ini dilakukan dengan memilih objek yang mempunyai berat teringan terlebih dahulu pada setiap langkahnya. Strategi ini bertujuan untuk mencoba memaksimalkan keuntungan dengan memasukkan objek sebanyak-banyaknya ke dalam kantong.

3. *Greedy by density*

Strategi *greedy by density* ini dilakukan dengan memilih objek yang mempunyai nilai p_i/w_i terbesar pada setiap langkahnya. Strategi ini bertujuan untuk mencoba memaksimalkan keuntungan dengan memilih objek yang mempunyai keuntungan per objek terbesar.

Sebagai contoh, diberikan 4 buah objek dengan nilai sebagai berikut:

$$w_1 = 6; p_1 = 12$$

$$w_3 = 10; p_3 = 50$$

$$w_2 = 5; p_2 = 15$$

$$w_4 = 5; p_4 = 10$$

dengan kapasitas *knapsack* $K = 16$, maka contoh persoalan ini dapat diselesaikan sebagai berikut:

Tabel 2.1 Tabel Strategi *Greedy*

Properti Objek				<i>Greedy by</i>			Solusi Optimal
i	w_i	p_i	p_i/w_i	<i>profit</i>	<i>weight</i>	<i>density</i>	
1	6	12	2	0	1	0	0
2	5	15	3	1	1	1	1
3	10	50	5	1	0	1	1
4	5	10	2	0	1	0	0
Total bobot				15	16	15	15
Total Keuntungan				65	37	65	65

Dari tabel di atas, didapat solusi optimal didapat dari persoalan ini diberikan oleh *greedy by density* dan *greedy by profit* dengan total keuntungan yang dapat diperoleh adalah 65.

Namun, tidak semua persoalan *knapsack* dapat diselesaikan dengan strategi *greedy*. Hal ini dibuktikan dengan contoh persoalan berikut:

Diberikan 6 buah objek sebagai berikut:

$$w_1 = 100; p_1 = 40$$

$$w_4 = 20; p_4 = 4$$

$$w_2 = 50; p_2 = 35$$

$$w_5 = 10; p_5 = 10$$

$$w_3 = 45; p_3 = 18$$

$$w_6 = 5; p_6 = 2$$

dengan kapasitas *knapsack* $K = 100$, maka penyelesaian persoalan ini dengan algoritma *greedy* sebagai berikut:

Tabel 2.2 Tabel Strategi *Greedy* (2)

Properti Objek				<i>Greedy by</i>			Solusi Optimal
i	w_i	p_i	p_i/w_i	<i>profit</i>	<i>weight</i>	<i>density</i>	
1	100	40	0,4	1	0	0	0
2	50	35	0,7	0	0	1	1
3	45	18	0,4	0	1	0	1
4	20	4	0,2	0	1	1	0
5	10	10	1,0	0	1	1	0
6	5	2	0,4	0	1	1	0
Total bobot				100	80	85	100
Total Keuntungan				40	34	51	55

Dari table di atas, ketiga strategi *greedy* gagal untuk memberikan solusi optimum yang diinginkan. Hal ini membuktikan bahwa pemilihan objek dengan salah satu dari

ketiga strategi *greedy* yang ada tidak menjamin akan memperoleh solusi yang optimal.

Selain permasalahan yang ternyata tidak semua permasalahan optimasi dapat diselesaikan dengan strategi *greedy*, strategi *greedy* juga sulit untuk melakukan pembuktian kebenaran. Justru, hal yang lebih mudah dilakukan adalah untuk menyangkal bahwa strategi *greedy* yang dibuat ternyata tidak menghasilkan solusi optimum dengan memberikan satu saja *counterexample* (contoh kasus yang membuktikan bahwa strategi yang dibuat tidak menghasilkan solusi optimum).

III. PENENTUAN BARANG KULAK DENGAN ALGORITMA GREEDY UNTUK MEMPEROLEH PROFIT MAKSIMAL

Penentuan barang yang akan dikulak dan jumlah barangnya tentu menjadi hal yang harus diperhatikan oleh pemilik toko kelontong ketika ingin melakukan kegiatan kulak. Hal ini karena jika salah dalam mengambil keputusan pembelian, maka pemilik toko dapat berpotensi rugi atau memperoleh keuntungan yang tidak maksimal. Penentuan barang ini dapat diselesaikan dengan strategi *greedy* yang telah dibahas. Sebelum menentukan strategi *greedy* yang akan dibuat, maka diperlukan beberapa data yang harus dimiliki oleh pemilik toko kelontong. Data yang harus dimiliki antara lain daftar nama barang, harga beli barang (membeli pada distributor), profit yang ditentukan tiap barangnya, stok awal barang, jumlah barang terjual selama periode waktu tertentu, persentase barang terjual dibanding dengan stok awal, jumlah peminat barang, serta modal yang dimiliki untuk membeli barang. Untuk jumlah peminat maksudnya adalah ada berapa banyak orang yang menanyakan ketersediaan suatu barang selama periode tertentu namun barang yang dimaksud stoknya sudah tidak ada di toko. Kemudian, untuk periode waktu dari data yang akan dibuat dapat menggunakan patokan dari dilakukannya kegiatan kulak sebelumnya hingga kegiatan kulak yang akan dilakukan sekarang, dapat berupa data mingguan, atau mungkin bulanan.

Setelah memiliki data-data yang diperlukan, maka perlu ditentukan terlebih dahulu elemen-elemen penyusun algoritma *greedy* serta formulasi dan *constraint* yang dibuat.

- Himpunan kandidat: himpunan barang yang ada di toko kelontong serta jumlah maksimal pembeliannya dengan jumlah bernilai lebih besar sama dengan 0.
- Himpunan solusi: barang-barang di toko kelontong yang terpilih beserta jumlah barang yang harus dibeli.
- Fungsi seleksi: memilih barang dengan profit tertinggi pada setiap langkahnya.
- Fungsi kelayakan: memeriksa apakah barang dengan jumlahnya yang baru dipilih apabila dijumlahkan dengan semua barang yang telah ada di himpunan solusi sebelumnya tidak melebihi modal uang yang dimiliki oleh pemilik toko kelontong. Modal uang sendiri akan ditentukan nantinya oleh pemilik toko kelontong.

- Fungsi objektif: memiliki profit yang maksimum pada setiap langkah perhitungannya sesuai dengan barang yang dipilih dan jumlah yang ditentukan.

Formulasi dan *constraint* untuk persoalan ini adalah sebagai berikut:

$$\text{Maksimasi } F = \sum_{i=1}^n \text{profit}_i \times \text{jumlahBarang}_i$$

dengan penentuan jumlah barang:

$$\text{jumlahBarang} \leq (\max(2 \times \text{terjual} - \text{stokAwal}, 0) + \text{peminat})$$

dan kendala (*constraint*):

$$\left(\sum_{i=1}^n \text{hargaBarang}_i \times \text{jumlahBarang}_i \right) \leq \text{modalAwal}$$

yang dalam hal ini, $i = 1, 2, 3, 4, \dots, n$

Penentuan jumlahBarang pada rumus tersebut memiliki tujuan untuk membatasi barang yang akan dibeli dengan menggunakan patokan bahwa kemungkinan barang terjual kedepannya akan mengikuti jumlah barang terjual sebelumnya dan ditambah peminatnya jika ada, sehingga jika ternyata stok awal masih lebih banyak dibanding barang terjual, maka tidak perlu melakukan pembelian barang.

Setelah menentukan elemen-elemen penyusun algoritma *greedy*, formulasi, serta *constraint* yang dimiliki, maka strategi ini dapat diimplementasikan pada persoalan ini.

A. Implementasi Algoritma Greedy pada Persoalan

Pada persoalan *integer knapsack problem* yang dibahas pada bagian sebelumnya, dapat dilihat bahwa strategi *greedy* yang dibuat tidak hanya berdasarkan satu hal saja. Pada *integer knapsack problem*, digunakan tiga pendekatan strategi yang tujuannya tetap sama, yaitu untuk memperoleh keuntungan yang maksimal. Serupa dengan *integer knapsack problem*, persoalan barang kulak ini juga dapat diselesaikan dengan beberapa strategi *greedy*. Beberapa strategi *greedy* yang dapat diimplementasikan yaitu:

- *Greedy by profit*

Strategi ini didasarkan pada kemungkinan bahwa semakin besar profit yang dihasilkan pada suatu barang, maka nantinya juga akan menghasilkan keuntungan yang maksimal juga. Strategi ini akan memilih barang dengan profit terbesar pada setiap langkahnya dan mengambil jumlah barang semaksimal mungkin juga kemudian memasukkannya ke dalam himpunan solusi hingga mencapai *constraint* yang telah ditetapkan.

- *Greedy by sold and interest*

Strategi ini didasarkan pada pengambilan total barang terjual ditambah peminat terbanyak pada setiap langkahnya. Hal ini karena dengan mengulak barang yang banyak terjual dan banyak diminati orang, maka

artinya semakin besar peluang barang tersebut akan cepat habis juga ketika dijual dan tentunya akan memberikan keuntungan yang tidak sedikit pula jika dijumlahkan. Barang dengan total penjualan dan peminat terbesar akan dimasukkan ke himpunan solusi hingga mencapai *constraint* yang telah ditetapkan.

- *Greedy by sold and interest percentage*

Strategi ini hampir sama dengan strategi sebelumnya, namun yang membedakannya adalah strategi ini menggunakan perbandingan total barang terjual ditambah peminat dibanding dengan stok awal barang di toko kelontong. Dengan strategi ini, diharapkan barang yang dikulak nantinya merupakan barang yang benar-benar laku di toko kelontong dan dapat menghasilkan profit yang banyak jika dijumlahkan. Strategi ini akan memasukkan barang dengan persentase terbesar pada setiap langkahnya ke dalam himpunan solusi hingga mencapai *constraint* yang telah ditetapkan.

Itu adalah beberapa strategi *greedy* yang mungkin untuk diterapkan pada persoalan ini. Semua strategi tentunya sama-sama memiliki tujuan untuk meraih profit terbesar dari kegiatan kulak yang akan dilakukan oleh pemilik toko kelontong dengan modal tertentu.

B. Data Sampel

Implementasi dari strategi *greedy* yang telah dibuat tentu memerlukan data sampel yang ada. Penulis telah menyediakan data sampel yang dibuat sendiri oleh penulis dengan beberapa data seperti harga barang, serta profit yang berasal dari situs milik [Kemendagri](#) dan situs [Distributor Sembako Murah](#) serta survey harga secara langsung oleh penulis di toko kelontong yang berada di dekat tempat tinggal penulis. Sedangkan, ada beberapa data juga yang murni buatan penulis seperti stok awal barang, jumlah barang terjual, dan jumlah peminat barang ketika barang tidak tersedia di toko kelontong. Asumsi data yang diambil adalah data penjualan selama satu bulan yang lalu (kegiatan kulak dilakukan tiap bulan). Barang yang disediakan pada data sampel juga terbatas di jumlahnya, yaitu sebanyak 20 jenis barang berbeda yang ada di toko kelontong. Berikut adalah data yang telah dibuat oleh penulis:

Tabel 3.1 Data Sampel Penjualan di Toko Kelontong

No	Nama Barang	Harga Beli*	Profit*	Stok Awal	Terjual	Peminat
1	Indomie Goreng	2.400	600	100	100	20
2	Gula pasir 1 kg	12.000	1.500	20	15	0
3	Minyak Goreng Tropical 1 Lite	11.700	2.300	20	18	0
4	So Klin Detergen	875	125	100	90	0
5	Sabun Cuci Sunlight	14.500	1.500	30	30	0

6	Sabun Batang Lifebuoy	3.000	500	50	20	0
7	Sabun Cair Lifebuoy	23.500	1.500	15	8	0
8	Sampo Sunslik Sachet	740	260	100	100	10
9	Sampo Sunslik Botol	14.000	1.000	20	20	0
10	Pasta Gigi Pepsodent	4.500	1.000	20	10	0
11	Biskuit Good Time	920	580	50	46	0
12	Wafer Cokelat Superstar	920	580	50	20	0
13	Wafer Cokelat Beng Beng	1.600	400	50	50	5
14	Lotion Anti Nyamuk Soffel	900	100	100	50	0
15	Aqua Botol 600 ml	1.700	800	20	15	0
16	Indomie Kuah Soto	2.400	600	100	85	0
17	Garam Kemasan 500 gr	5.400	1.100	50	38	0
18	Beras Kemasan 1 kg	12.200	1.100	20	20	2
19	Tepung Terigu Segitiga Biru 1 kg	13.900	1.100	20	5	0
20	Kopi Sachet Kapal Api	1.000	500	50	50	10

*harga tertera merupakan harga per barang

C. Perhitungan dengan Strategi Greedy

Setelah mendapatkan data yang diperlukan, maka perhitungan dengan strategi *greedy* yang telah ditentukan dapat dilakukan. Sebelum melakukan itu, kita harus menentukan terlebih dahulu jumlah setiap barang yang harus dibeli sesuai dengan rumus yang telah dibuat. Berikut adalah table jumlah barang maksimal yang harus dibeli:

Tabel 3.2 Jumlah Masing-Masing Barang yang Harus Dibeli

No	Nama Barang	Stok Awal	Terjual	Peminat	Jumlah Barang
1	Indomie Goreng	100	100	20	120
2	Gula pasir 1 kg	20	15	0	10
3	Minyak Goreng Tropical 1 Lite	20	18	0	16

4	So Klin Detergen	100	90	0	80
5	Sabun Cuci Sunlight	30	30	0	30
6	Sabun Batang Lifebuoy	50	20	0	0
7	Sabun Cair Lifebuoy	15	8	0	1
8	Sampo Sunsik Sachet	100	100	10	110
9	Sampo Sunsilk Botol	20	20	0	20
10	Pasta Gigi Pepsodent	20	10	0	0
11	Biskuit Good Time	50	46	0	42
12	Wafer Cokelat Superstar	50	20	0	0
13	Wafer Cokelat Beng Beng	50	50	5	55
14	Lotion Anti Nyamuk Soffel	100	50	0	0
15	Aqua Botol 600 ml	20	15	0	10
16	Indomie Kuah Soto	100	85	0	70
17	Garam Kemasan 500 gr	50	38	0	26
18	Beras Kemasan 1 kg	20	20	2	22
19	Tepung Terigu Segitiga Biru 1 kg	20	5	0	0
20	Kopi Sachet Kapal Api	50	50	10	60

Setelah mengetahui jumlah barang yang harus dibeli atau batas maksimal pembelian barang (jumlah barang dapat berkurang apabila modal awal yang dimiliki tidak cukup untuk membeli seluruhnya) maka perhitungan dapat dilakukan. Perhitungan ini akan menggunakan modal awal yaitu sebesar Rp1.500.000 (Nilai K = 1.500.000).

Berikut adalah hasil perhitungan apabila menggunakan strategi *greedy by profit*:

Tabel 3.3 Hasil Greedy by Profit

No	Nama Barang	Harga Beli*	Profit*	Greedy by Profit (jumlah Barang yang dibeli)	Total Profit	Total Harga
1	Indomie Goreng	2.400	600	11	6.600	26.400
2	Gula pasir 1 kg	12.000	1.500	10	15.000	120.000
3	Minyak Goreng Tropical 1 Lite	11.700	2.300	16	36.800	187.200
4	So Klin Detergen	875	125	0	0	0
5	Sabun Cuci Sunlight	14.500	1.500	30	45.000	435.000
6	Sabun Batang Lifebuoy	3.000	500	0	0	0
7	Sabun Cair Lifebuoy	23.500	1.500	1	1.500	23.500

8	Sampo Sunsik Sachet	740	260	0	0	0
9	Sampo Sunsik Botol	14.000	1.000	20	20.000	280.000
10	Pasta Gigi Pepsodent	4.500	1.000	0	0	0
11	Biskuit Good Time	920	580	2	1.160	1.840
12	Wafer Cokelat Superstar	920	580	0	0	0
13	Wafer Cokelat Beng Beng	1.600	400	0	0	0
14	Lotion Anti Nyamuk Soffel	900	100	0	0	0
15	Aqua Botol 600 ml	1.700	800	10	8.000	17.000
16	Indomie Kuah Soto	2.400	600	0	0	0
17	Garam Kemasan 500 gr	5.400	1.100	26	28.600	140.400
18	Beras Kemasan 1 kg	12.200	1.100	22	24.200	268.400
19	Tepung Terigu Segitiga Biru 1 kg	13.900	1.100	0	0	0
20	Kopi Sachet Kapal Api	1.000	500	0	0	0
Total					186.860	1.499.740

*harga tertera merupakan harga per barang

Dari tabel di atas, dapat dilihat, dari modal awal senilai Rp1.500.000 dapat dibelanjakan barang dengan total Rp1.499.740 dengan rincian barang yang dibeli dan jumlahnya ada pada tabel. Jumlah barang yang harus dibeli ada pada kolom *Greedy by Profit* yang artinya ada beberapa barang yang tidak dapat dibeli semuanya karena keterbatasan modal awal. Dengan *greedy by profit* ini didapatkan profit total sebesar Rp186.860.

Kemudian, berikut adalah hasil perhitungan apabila menggunakan strategi *greedy by sold and interest*:

Tabel 3.4 Hasil Greedy by Sold and Interest

No	Nama Barang	Harga Beli*	Profit *	Total Terjual dan Peminat	Greedy by Sold and Interest	Total Profit	Total Harga
1	Indomie Goreng	2.400	600	120	120	72.000	288.000
2	Gula pasir 1 kg	12.000	1.500	15	0	0	0
3	Minyak Goreng Tropical 1 Lite	11.700	2.300	18	0	0	0
4	So Klin Detergen	875	125	90	80	10.000	70.000
5	Sabun Cuci	14.500	1.500	30	30	45.000	435.000

	Sunlight						
6	Sabun Batang Lifebuoy	3.000	500	20	0	0	0
7	Sabun Cair Lifebuoy	23.500	1.500	8	0	0	0
8	Sampo Sunsik Sachet	740	260	110	110	28.600	81.400
9	Sampo Sunsilk Botol	14.000	1.000	20	0	0	0
10	Pasta Gigi Pepsodent	4.500	1.000	10	0	0	0
11	Biskuit Good Time	920	580	46	42	24.360	38.640
12	Wafer Cokelat Superstar	920	580	20	0	0	0
13	Wafer Cokelat Beng Beng	1.600	400	55	55	22.000	88.000
14	Lotion Anti Nyamuk Soffel	900	100	50	0	0	0
15	Aqua Botol 600 ml	1.700	800	15	5	4.000	8.500
16	Indomie Kuah Soto	2.400	600	85	70	42.000	168.000
17	Garam Kemasan 500 gr	5.400	1.100	38	26	28.600	140.400
18	Beras Kemasan 1 kg	12.200	1.100	22	10	11.000	122.000
19	Tepung Terigu Segitiga Biru 1 kg	13.900	1.100	5	0	0	0
20	Kopi Sachet Kapal Api	1.000	500	60	60	30.000	60.000
Total						317.560	1.499.940

*harga tertera merupakan harga per barang

Dari tabel di atas, dapat dilihat, dari modal awal senilai Rp1.500.000 dapat dibelanjakan barang dengan total Rp1.499.940 dengan rincian barang yang dibeli dan jumlahnya ada pada tabel. Jumlah barang yang harus dibeli ada pada kolom *Greedy by Sold and Interest* yang artinya ada beberapa barang yang tidak dapat dibeli semuanya karena keterbatasan modal awal. Dengan *greedy by sold and interest* ini didapatkan profit total sebesar Rp317.560.

Kemudian, strategi *greedy* yang terakhir adalah strategi *greedy by sold and interest percentage*. Berikut adalah hasil perhitungannya:

Tabel 3.4 Hasil *Greedy by Sold and Interest Percentage*

No	Nama Barang	Harga Beli*	Profit *	Persentase Terjual dan Peminat (%)	Greedy by Sold and Interest Percentage	Total Profit	Total Harga
1	Indomie Goreng	2.400	600	120	120	72.000	288.000

2	Gula pasir 1 kg	12.000	1.500	75	0	0	0
3	Minyak Goreng Tropical 1 Lite	11.700	2.300	90	0	0	0
4	So Klin Detergen	875	125	90	0	0	0
5	Sabun Cuci Sunlight	14.500	1.500	100	30	45.000	435.000
6	Sabun Batang Lifebuoy	3.000	500	40	0	0	0
7	Sabun Cair Lifebuoy	23.500	1.500	53,3	0	0	0
8	Sampo Sunsik Sachet	740	260	110	110	28.600	81.400
9	Sampo Sunsilk Botol	14.000	1.000	100	19	19.000	266.000
10	Pasta Gigi Pepsodent	4.500	1.000	50	0	0	0
11	Biskuit Good Time	920	580	92	14	8.120	12.880
12	Wafer Cokelat Superstar	920	580	40	0	0	0
13	Wafer Cokelat Beng Beng	1.600	400	110	55	22.000	88.000
14	Lotion Anti Nyamuk Soffel	900	100	50	0	0	0
15	Aqua Botol 600 ml	1.700	800	75	0	0	0
16	Indomie Kuah Soto	2.400	600	85	0	0	0
17	Garam Kemasan 500 gr	5.400	1.100	76	0	0	0
18	Beras Kemasan 1 kg	12.200	1.100	110	22	24.200	268.400
19	Tepung Terigu Segitiga Biru 1 kg	13.900	1.100	25	0	0	0
20	Kopi Sachet Kapal Api	1.000	500	120	60	30.000	60.000
Total						248.920	1.499.680

*harga tertera merupakan harga per barang

Dari tabel di atas, dapat dilihat, dari modal awal senilai Rp1.500.000 dapat dibelanjakan barang dengan total Rp1.499.680 dengan rincian barang yang dibeli dan jumlahnya ada pada tabel. Jumlah barang yang harus dibeli ada pada kolom *Greedy by Sold and Interest Percentage* yang artinya ada beberapa barang yang tidak dapat dibeli semuanya karena keterbatasan modal awal. Dengan *greedy by sold and interest percentage* ini didapatkan profit total sebesar Rp248.920.

Apabila ketiga strategi tersebut dibandingkan, maka akan terlihat sebagai berikut:

Tabel 3.5 Perbandingan Ketiga Strategi *Greedy*

Strategi	Total Belanja	Total Profit
<i>Greedy by Profit</i>	Rp1.499.740	Rp186.860
<i>Greedy by Sold and Interest</i>	Rp1.499.940	Rp317.560
<i>Greedy by Sold and Interest Percentage</i>	Rp1.499.680	Rp248.920

Dari perbandingan yang telah dibuat, terlihat bahwa strategi *greedy by sold and interest* dapat menghasilkan profit lebih besar dibanding dengan strategi yang lain. Namun perlu diingat, bahwa ada kemungkinan dengan uji kasus yang lain strategi *greedy* yang lain juga dapat menghasilkan solusi yang optimal sehingga dalam penentuan barang yang akan dikulak dan jumlah barangnya dapat dilakukan dengan ketiga strategi yang telah dibuat dan dibandingkan mana yang menghasilkan profit lebih maksimal.

IV. KESIMPULAN

Kegiatan kulak yang dilakukan oleh pemilik toko kelontong dapat memanfaatkan algoritma *greedy* dengan tujuan untuk memaksimalkan profit penjualan dari barang yang dikulak nantinya. Algoritma *greedy* terkenal dapat menyelesaikan permasalahan yang berkaitan dengan optimasi, salah satunya adalah kegiatan kulak ini. Sebelum menggunakan algoritma *greedy*, diperlukan beberapa data yang nantinya akan menjadi elemen-elemen penyusun algoritma ini. Data yang diperlukan yaitu nama barang yang akan dikulak, harga beli barang, profit tiap barang, stok awal barang sebelum kulak, jumlah barang terjual, serta peminat barang ketika barang sedang tidak tersedia di toko.

Dengan data yang diperoleh, maka strategi *greedy* dapat dilakukan dengan modal awal yang dapat ditentukan sendiri oleh pemilik tok kelontong, dalam contoh kasus ditetapkan nilai modal awal yaitu senilai Rp1.500.000. Strategi *greedy* yang dibuat yaitu *greedy by profit*, *greedy by sold and interest*, dan *greedy by sold and interest percentage* dan didapat bahwa dengan strategi *greedy by sold and interest* diperoleh keuntungan terbesar dibanding strategi yang lain. Namun, perlu diingat, perbedaan data dan modal awal bisa saja mengubah keuntungan masing-masing strategi sehingga sangat disarankan untuk tetap menggunakan ketiga strategi ketika akan melakukan kegiatan kulak dan membandingkan mana yang menghasilkan profit paling besar.

VIDEO LINK DI YOUTUBE

<https://youtu.be/kn3UMAazPU4>

UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang maha Esa atas diberinya kesempatan, kemudahan, dan kelancaran bagi penulis untuk mengerjakan makalah ini. Penulis mengucapkan terima kasih kepada pihak-pihak yang telah membantu penulis dalam menyelesaikan makalah ini. Tak lupa, penulis juga menyampaikan terima kasih kepada seluruh dosen pengampu mata kuliah Strategi Algoritma, utamanya Bapak Rinaldi Munir sebagai dosen pengampu kelas K3 yang telah bersedia untuk menyampaikan ilmunya kepada penulis.

REFERENSI

- [1] Munir, Rinaldi, 2021. Algoritma Greedy (Bagian 1). Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf) pada 21 Mei 2022.
- [2] Munir, Rinaldi, 2021. Algoritma Greedy (Bagian 2). Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag2.pdf) pada 21 Mei 2022.
- [3] Munir, Rinaldi, 2022. Algoritma Greedy (Bagian 3). Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-\(2022\)-Bag3.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-(2022)-Bag3.pdf) pada 21 Mei 2022.
- [4] Gozali, William, and A. F. Aji. 2019. Pemrograman Kompetitif Dasar Versi 1.9. Jakarta: CV. Nulisbuku Jendela Dunia.
- [5] Wahyu Octian, Ardito. 2022. Apa Itu Toko Kelontong? Inilah Pengertian dan Jenis-Jenisnya. Diakses dari <https://mitra.bukalapak.com/artikel/apa-itu-toko-kelontong-116025> pada 20 Mei 2022.

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, 22 Mei 2022



Haider Ihzaulhaq
13520150