

# Aplikasi Algoritma Branch and Bound dalam Penentuan Menu pada Rumah Makan sekitar Institut Teknologi Bandung

Cynthia Rusadi / 13519118  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung  
13519118@std.stei.itb.ac.id

**Abstrak**—Pangan adalah salah satu kebutuhan pokok manusia dan merupakan hal yang umum jika setiap orang mengalami kebingungan saat berhadapan dengan pangan, dengan berbagai alasan. Salah satunya yang berdampak adalah mahasiswa merantau, yang terbatas secara keuangan dan harus memiliki asupan yang cukup dan baik, khususnya mahasiswa Institut Teknologi Bandung, dan masalah tersebut dapat diselesaikan dengan menggunakan algoritma *branch and bound*.

**Keywords**—mahasiswa, *branch and bound*, *integer knapsack problem*, *anggaran*

## I. PENDAHULUAN

Pangan merupakan salah satu kebutuhan pokok manusia dan terdapat beberapa cara untuk mendapatkan kebutuhan ini, yaitu dengan bertanam, berburu, beternak, memproduksi, atau dengan cara instan. Cara instan yang paling umum dilakukan pada zaman ini adalah dengan membeli langsung di sebuah rumah makan, yang memiliki banyak variasi dan jumlah, sehingga dapat dikonsumsi bersamaan dengan keluarga atau kawan.

Sudah menjadi hal yang umum bahwa setiap orang akan mengalami kebingungan ketika ingin memesan makanan di rumah makan berdasarkan menunya, maupun itu karena tidak ada preferensi, ingin mencoba banyak jenis makanan, pilihan yang terbatas, anggaran yang terbatas, atau yang lain-lainnya. Sebagai mahasiswa Institut Teknologi Bandung yang merantau dan berada di lingkungan yang berisi mahasiswa yang merantau dari kota asal, fakta bahwa memilih menu pada rumah makan merupakan salah satu tantangan karena harus memperhatikan anggaran, dengan kondisi harus mengonsumsi makanan yang bernutrisi, merupakan sebuah tantangan.

Salah satu solusi untuk mengatasi tantangan tersebut adalah dengan menggunakan algoritma *branch and bound*, yaitu dengan menentukan jumlah makanan yang akan dibeli dan anggaran yang tersedia. Dengan menggunakan salah satu contoh persoalan algoritma *branch and bound*, yaitu *integer knapsack problem*, akan didapatkan solusi paling optimal agar mahasiswa dapat menghemat dan mengonsumsi makanan yang cukup untuk menjalani kehidupan sehari-hari.

## II. TEORI DASAR

### A. *Branch and Bound*

Algoritma *Branch and Bound* merupakan salah satu yang digunakan untuk pencarian jalur dan digunakan untuk persoalan optimisasi. Optimisasi yang dilakukan adalah dengan meminimalkan atau memaksimalkan sebuah fungsi objektif, dengan syaratnya adalah tidak melanggar Batasan dari persoalan tersebut. *Branch and Bound* merupakan kombinasi algoritma *Breadth First Search* (BFS), yang melakukan pencarian simpul berikutnya berdasarkan urutan pembangkitan, dan *least cost search*. Algoritma ini memiliki 2 prinsip, yaitu:

1. Melakukan perhitungan secara rekursif, dengan melakukan pemecahan masalah ke masalah yang lebih kecil, dengan tetap memperhitungkan nilai terbaiknya. Proses ini dapat dinamakan sebagai *branching*
2. Jika *branching* diterapkan secara sendirian, hasilnya akan tetap mencari setiap kemungkinan yang ada. Untuk meningkatkan performanya, algoritma ini akan melakukan pencatatan biaya minimum sebagai *bound* untuk setiap perhitungannya sehingga untuk calon hasil jawaban yang diperkirakan akan melebihi *bound*, akan dibuang karena tidak memungkinkan untuk mencapai nilai terbaik

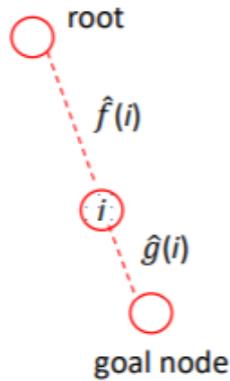
Persoalan yang dapat diselesaikan oleh algoritma *branch and bound* merupakan persoalan optimisasi, yang berarti untuk setiap simpul yang terdapat pada suatu pohon ruang-status, dibutuhkan suatu cara untuk menentukan batas (*bound*) nilai terbaik fungsi objektif untuk setiap solusi yang memungkinkan, dan juga dengan menambahkan komponen pada solusi-solusi sementara tersebut yang direpresentasikan oleh simpulnya. Nilainya merupakan solusi yang terbaik.

Untuk menentukan langkah selanjutnya yang akan diambil, algoritma *branch and bound* memperhitungkan biaya (*cost*) berdasarkan persoalannya untuk setiap simpulnya. Biaya untuk setiap simpulnya merupakan nilai taksiran lintasan termurah ke simpul status tujuan yang melalui simpul status  $i$ , dilambangkan sebagai  $\hat{c}(i)$ . Simpul selanjutnya yang akan di-

*expand* tidak lagi berdasarkan urutan pembangkitannya, melainkan berdasarkan simpul yang memiliki biaya terkecil, atau dapat dikatakan juga sebagai *least cost search*. Dalam penentuan biaya dari simpul hidup, pada umumnya letak simpul solusinya tidak diketahui, maka dari itu  $\hat{c}(i)$  dihitung secara heuristik, atau dapat dikatakan sebagai batas bawah dari ongkos pencarian solusi dari status  $i$ . Biaya untuk setiap simpul pada umumnya adalah

$$\hat{c}(i) = \hat{g}(i) + \hat{h}(i) \tag{1}$$

dengan  $\hat{h}(i)$  adalah ongkos untuk mencapai simpul  $i$  dari akar dan  $\hat{g}(i)$  adalah ongkos untuk mencapai simpul tujuan dari simpul  $i$ .

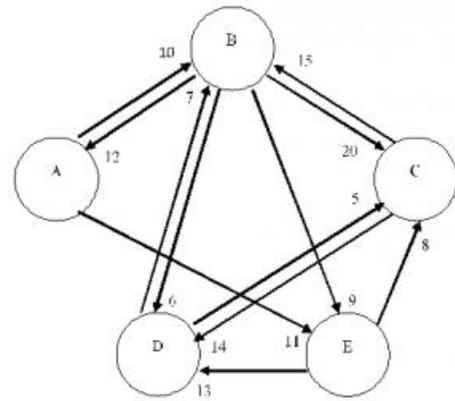


Gambar 2.1. Ilustrasi Biaya dari Simpul Hidup  
Sumber: [2]

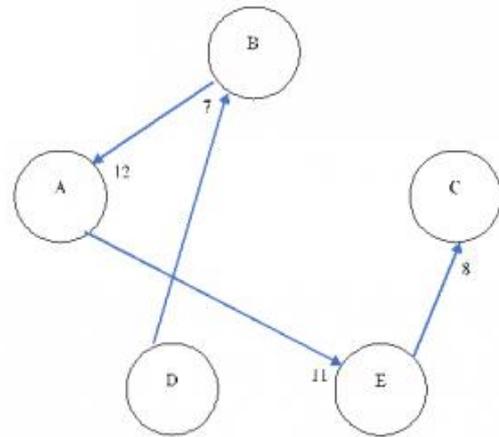
Langkah-langkah yang dilakukan untuk menentukan algoritma global *branch and bound* adalah sebagai berikut:

1. Memasukkan simpul akar ke dalam antrian Q. Jika simpul akar tersebut merupakan simpul solusinya, solusi telah ditemukan dan proses berhenti
2. Jika Q kosong, tidak ada solusi dan proses berhenti
3. Jika Q tidak kosong, melakukan pemilihan dari antrian Q simpul  $i$  yang memiliki  $\hat{c}(i)$  terkecil. Jika terdapat beberapa simpul  $i$  yang memenuhi, pilihlah salah satu simpul secara acak
4. Jika simpul  $i$  merupakan simpul solusi, solusi telah ditemukan dan proses berhenti. Jika simpul  $i$  bukan merupakan simpul solusi, bangkitkan semua anak dari simpul tersebut dan jika  $i$  tidak mempunyai anak, kembali ke langkah 2
5. Untuk setiap anak  $j$  dari simpul  $i$ , lakukan kalkulasi untuk  $\hat{c}(j)$  dan masukkan semua anak-anaknya ke dalam antrian Q
6. Kembali ke langkah 2

Salah satu contoh penyelesaian dari persoalannya adalah seperti gambar di bawah, dengan simpul awal D dan simpul tujuan C.



Gambar 2.2. Contoh Persoalan Branch and Bound  
Sumber: [6]



Gambar 2.3. Contoh Penyelesaian dari Gambar 2.3.  
Sumber: [6]

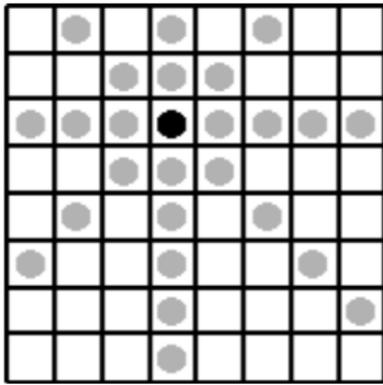
### B. Persoalan Branch and Bound

Terdapat beberapa persoalan yang dapat ditemukan solusinya dengan menggunakan algoritma *branch and bound*, yaitu:

#### 1. Persoalan N-Ratu (*The N-Queens Problem*)

Merupakan persoalan sebuah papan permainan yang berukuran  $N \times N$  dan  $N$  buah ratu. Ratu-ratu tersebut harus ditempatkan pada petak papan tersebut sedemikian rupa agar tidak ada dua ratu atau lebih yang terletak pada baris, kolom, dan diagonal yang sama. Simpul yang memiliki nilai biaya terkecil adalah simpul hidup yang menjadi simpul *expand*, dan untuk setiap simpul X, nilai biayanya dapat berupa:

- a. Jumlah simpul yang terdapat di dalam upapohon X dan perlu dibangkitkan sebelum ditemukan simpul solusinya
- b. Panjang lintasan dari simpul X sampai ke simpul solusi terdekat



Gambar 2.4. Persoalan N-Ratu  
Sumber: [2]

2. Permainan 15-Puzzle

Adalah persoalan yang menentukan urutan langkah dari pergeseran sebuah kotak kosong untuk dijadikan sebuah puzzle dengan posisi dari setiap nomornya sesuai dengan posisi pada solusi akhirnya. Perhitungan biaya yang dilakukan pada permainan 15-Puzzle adalah

$$\hat{c}(P) = f(P) + \hat{g}(P) \quad (2)$$

dengan  $f(P)$  adalah panjang lintasan dari simpul akar ke  $P$  dan  $\hat{g}(P)$  adalah taksiran panjang lintasan terpendek dari  $P$  sampai ke simpul solusi pada upapohon yang akarnya adalah  $P$ , atau jumlah ubin tidak kosong yang posisinya tidak sesuai dengan posisi pada solusi akhirnya.

1	3	4	15
2		5	12
7	6	11	14
8	9	10	13

(a) Susunan awal

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

(b) Susunan akhir

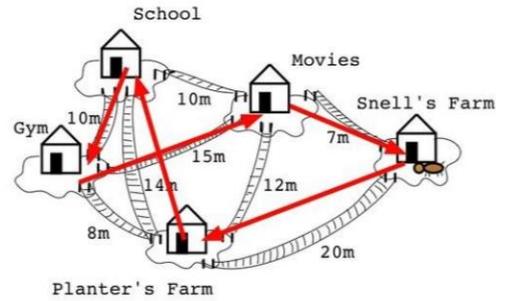
Gambar 2.5. Permainan 15-Puzzle  
Sumber: [2]

3. Travelling Salesperson Problem (TSP)

Persoalan dari TSP adalah suatu kondisi di mana diberikan sebuah  $n$  kota (simpul) dan diketahui jarak (bobot) antarkota tersebut. Perlu ditemukan perjalanan (*tour*) dengan jarak terpendek yang dapat dilalui oleh seorang pedagang, dengan syarat pedagang tersebut harus melalui setiap kota tepat hanya sekali saja dan kembali ke kota asal keberangkatannya. *Cost* yang diperhitungkan untuk setiap simpulnya di dalam pohon ruang status menyatakan nilai batas bawah (*lower bound*) ongkos untuk mencapai simpul solusi dari simpul tersebut, dan dapat dihitung secara heuristik berdasarkan salah satu dari kedua cara ini:

- a. Matriks ongkos-tereduksi (*reduced cost matrix*) dari graf

b. Bobot minimum tur lengkap



Gambar 2.6. Travelling Salesperson Problem  
Sumber: [3]

4. Assignment Problem

Merupakan persoalan di saat terdapat  $n$  orang dan  $n$  buah pekerjaan (*job*) dan masing-masing orang akan diberikan pekerjaan. Biaya untuk memerikan pekerjaan untuk setiap orang dinyatakan dalam sebuah matriks. Cara perhitungan untuk biaya (*lower bound*) setiap simpul hidup di dalam pohon ruang status bervariasi, seperti:

- a. Menggunakan matriks ongkos tereduksi
- b. Menjumlahkan nilai minimum untuk setiap baris pada matriks tersebut

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

Gambar 2.7. Assignment Problem  
Sumber: [4]

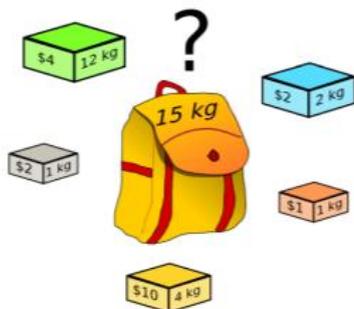
5. Integer Knapsack Problem

Merupakan persoalan dengan situasi ketika diberikan  $n$  buah objek dan sebuah *knapsack* dengan kapasitas bobot  $K$ . Untuk setiap objeknya, akan memiliki bobot (*weight*)  $w_i$  dan keuntungan (*profit*)  $p_i$ , dan perlu diperhatikan cara pemilihan objek-objek untuk dimasukkan ke dalam *knapsack* agar dapat memperoleh keuntungan yang maksimal, dengan batasannya adalah jumlah bobotnya tidak melebihi kapasitas *knapsack*. Persoalan ini dapat dikatakan sebagai persoalan maksimasi, atau mencari keuntungan maksimum, oleh karena itu biaya ada setiap simpul di pohon ruang status dinyatakan sebagai batas atas (*upper bound*) dari solusi optimal. Simpul yang di-*expand* adalah simpul hidup yang memiliki biaya terbesar. Perhitungan untuk biayanya adalah

$$\hat{c}(i) = F + (K - W)p_{i+1}/w_{i+1} \quad (3)$$

dengan  $F$  adalah total keuntungan yang sudah dicapai,  $K$  adalah bobot *knapsack*,  $W$  adalah total bobot

*knapsack* yang sudah terpakai, dan  $(p_{i+1}/w_{i+1})$  adalah rasio keuntungan per bobot objek yang tersisa untuk berikutnya.



Gambar 2.8. Integer Knapsack Problem  
Sumber: [4]

### III. APLIKASI ALGORITMA BRANCH AND BOUND

#### A. Rumah Makan di sekitar Insitut Teknologi Bandung

Sebelum menentukan menu, terdapat beberapa daftar rumah makan yang terdapat di sekitar Institut Teknologi Bandung, seperti:

##### 1. McDonald's Dago

McDonald's adalah salah satu rumah makan cepat saji terbesar di dunia yang diawali pada tahun 1955 di California, Amerika Serikat, dengan produk unggulannya merupakan Bigmac. Hingga saat ini, McDonald's sudah memiliki ribuan cabang yang tersebar di lebih dari 100 negara, dengan salah satunya adalah Indonesia. McDonald's pertama masuk ke Indonesia pada tahun 1991 dan pada tahun 2009, PT. Rekso Nasional Food (RNF) menandatangani *Master Franchise Agreement* dengan McDonald's International Property Company (MIPCO) yang memberikan izin untuk mengoperasikan semua restoran dengan brand McDonald's dan membuka restoran baru di seluruh Indonesia, dan salah satunya adalah McDonald's Dago.



Gambar 3.1. McDonald's Dago  
Sumber: <https://www.traveloka.com/id-restaurants/indonesia/detail/mcdonalds-simpang-dago-46016> diakses pada 7 Mei 2021, 20:59

##### 2. Ayam Geprek Crisbar Cisitu

Usaha ini telah dirintis sejak 27 Agustus 2017 oleh Hafizh dan Filbert. Saat itu Filbert mengajak Hafizh untuk mendirikan usaha dan Hafizh berinisiasi bahwa idenya berupa ayam geprek, yang berasal dari kota asalnya, Yogyakarta. Pada bulan pertamanya, kurang lebih 200-300 porsi Ayam Geprek Crisbar berhasil terjual dan hasilnya digunakan untuk membeli gerobak pertama. Pada bulan kedua, cabang pertamanya dibuka di Ganesha dan bulan ketiga dan keempat, cabang baru dibuka di Cisitu sebagai target strategis selanjutnya. Pada bulan kelima didirikan cabang baru di Cimahi, bulan keenam didirikan cabang baru di Universitas Telkom, dan cabang selanjutnya didirikan di Maranata dan Buah Batu pada bulan ketujuh. Saat ini pendapatan rata-rata per bulannya adalah sekitar 40-50 juta untuk per cabang.



Gambar 3.2. Ayam Geprek Crisbar Cisitu  
Sumber: <https://www.traveloka.com/id-restaurants/indonesia/detail/ayam-geprek-crisbar-cisitu-87666/photos> diakses pada 7 Mei 2021, 22:06

#### B. Menu Rumah Makan di sekitar Institut Teknologi Bandung

##### 1. McDonald's Dago

Berikut merupakan salah satu menu yang disediakan pada McDonald's Dago dan umum dipesan oleh mahasiswa.

<b>1 Big Mac®</b> Paket Hemat: Rp 41.210 A la Carte: Rp 30.000	<b>2 Double Cheeseburger</b> Paket Hemat: Rp 40.300 A la Carte: Rp 28.500	<b>3 McSpicy</b> Paket Hemat: Rp 44.545 A la Carte: Rp 31.304	<b>4 McChicken®</b> Paket Hemat: Rp 37.273 A la Carte: Rp 25.000	<b>5 McNuggets®</b> Paket Hemat: Rp 25.000 A la Carte: Rp 23.102
<b>Paket HEBAT Hemat Banget</b>				
<b>6 Cheeseburger</b> Paket HEBAT: Rp 30.300 A la Carte: Rp 22.373	<b>7 Cheeseburger with Egg</b> Paket HEBAT: Rp 34.000 A la Carte: Rp 23.000	<b>8 Cheeseburger Deluxe</b> Paket HEBAT: Rp 38.273 A la Carte: Rp 23.102	<b>9 Chicken Burger with Cheese</b> Paket HEBAT: Rp 30.000 A la Carte: Rp 21.810	
Paket sudah termasuk Medium French Fries dan Medium Drink Tambahkan 1pc Ayam Goreng McD hanya Rp15.450, setiap pembelian Paket.				

Gambar 3.4. Menu McDonald's Dago

Sumber:

<https://pergikuliner.com/restaurants/mcdonald-s-dago-atas/menus> diakses pada 7 Mei 2021, 22:51

2. Ayam Geprek Crisbar Cisitu

Ayam Geprek Crisbar Cisitu sangat populer di kalangan mahasiswa Institut Teknologi Bandung dan beberapa menu favoritnya adalah sebagai berikut.



Gambar 3.5. Menu Ayam Geprek Crisbar Cisitu  
 Sumber: <https://www.crispybakar.biz/our-menu> diakses pada 7 Mei 2021, 22:54



Gambar 3.6. Menu Ayam Geprek Crisbar Cisitu (2)  
 Sumber: <https://www.crispybakar.biz/our-menu> diakses pada 7 Mei 2021, 22:54

Big Mac	30.000	257
Double Cheeseburger	28.636	440
McSpicy	31.364	585
McChicken	25.000	350
McNuggets	23.182	396
Cheeseburger	22.273	300
Cheeseburger with Egg	23.636	323
Cheeseburger Deluxe	23.182	400
Chicken Burger with Cheese	21.818	413

Tabel 3.1. Daftar Nama, Harga, dan Kalori Menu McDonald's Dago

Dengan salah satu persoalannya adalah jika seorang mahasiswa ingin membeli makanan dari McDonald's Dago dengan anggarannya sebesar Rp100.000,00, dengan asumsi tidak memperhitungkan pajak, dengan tujuan ingin mentraktir temannya, tetapi berharap agar teman-temannya juga mendapatkan kalori yang tidak kurang maupun berlebih, maka solusinya adalah sebagai berikut:

```

    +=+=+=+=+= Menu McDonal's Dago +=+=+=+=+=
    1. Big Mac: Rp30000,00
    2. Double Cheeseburger: Rp28636,00
    3. McSpicy: Rp31364,00
    4. McChicken: Rp25000,00
    5. McNuggets: Rp23182,00
    6. Cheeseburger: Rp22273,00
    7. Cheeseburger with Egg: Rp23636,00
    8. Cheeseburger Deluxe: Rp23182,00
    9. Chicken Burger with Cheese: Rp21818,00
    Masukkan jumlah anggaran yang dimiliki: 100000

    Makanan yang akan dipesan:
    1. McSpicy
    1. McNuggets
    1. Cheeseburger Deluxe
    1. Chicken Burger with Cheese
    Total kalori: 1794
    Total harga: Rp99546
    
```

Gambar 3.7. Solusi untuk McDonald's Dago  
 Sumber: Dokumentasi pribadi

C. Contoh Penerapan Algoritma Branch and Bound Menggunakan Integer Knapsack Problem

Permasalahan ini dapat diselesaikan dengan menggunakan program sederhana, bahasa Python, dengan mengimplementasikan prinsip Integer Knapsack Problem.

1. McDonald's Dago

Berikut adalah daftar makanan, harga, beserta kalorinya, berdasarkan *a la carte*. Daftar harga berdasarkan gambar 3.4.

Nama	Harga (Rp)	Kalori (kcal)
------	------------	---------------

2. Ayam Geprek Crisbar Cisitu

Berdasarkan Gambar 3.5. dan Gambar 3.6., daftar nama dan harga dari menunya, berdasarkan paketnya, adalah

Nama	Harga (Rp)
Puass Crisbar	20000
Puass Sambal Matah	19000

Puass Sambal Mamah	19000
Puass Sambal Goang	19000
Puass Sambal Tomat	19000
Puass Manis	18000
Puass Keju	18000
Geprek Crispy	17000
Ayam Crispy	17000

Tabel 3.2. Daftar Nama dan Harga Menu Ayam Geprek Crisbar Cisitu

Jika terdapat sebuah permasalahan, yaitu seorang mahasiswa sedang dalam kondisi sangat lapar setelah berolahraga di Saraga, dan ingin memakan sebanyak mungkin, tetapi karena anggaran yang dimiliki adalah Rp55.000,00, maka mahasiswa tersebut membuat keputusan yaitu pergi ke Ayam Geprek Crisbar Cisitu dan solusinya adalah sebagai berikut

```

+=+=+=+= Menu Ayam Geprek Crisbar Cisitu +=+=+=+=
1. Puass Crisbar: Rp20000,00
2. Puass Sambal Matah: Rp19000,00
3. Puass Sambal Mamah: Rp19000,00
4. Puass Sambal Goang: Rp19000,00
5. Puass Sambal Tomat: Rp19000,00
6. Puass Manis: Rp18000,00
7. Puass Keju: Rp18000,00
8. Geprek Crispy: Rp17000,00
9. Ayam Crispy: Rp17000,00
Masukkan jumlah anggaran yang dimiliki: 55000

Makanan yang akan dipesan:
1. Puass Crisbar
1. Geprek Crispy
1. Ayam Crispy
Total makanan: 3
Total harga: Rp54000

```

Gambar 3.8. Solusi untuk Ayam Geprek Crisbar Cisitu  
Sumber: Dokumentasi pribadi

Kedua persoalan tersebut dapat diselesaikan dengan mudah dengan menggunakan program bahasa Python dan algoritmanya adalah seperti di gambar di bawah.

```

def knapSack(batas_harga, list_harga, list_cost, list_f):
    f = 0
    total_harga = 0 # w
    list_accepted = [0 for i in range (len(list_harga))]
    for i in range (len(list_harga)):
        cost_1 = f + ((batas_harga - total_harga) * list_cost[i])
        if batas_harga >= total_harga + list_harga[i]:
            cost_2 = (f + list_f[i]) + ((batas_harga - (total_harga + list_harga[i])) * list_cost[i])
            if cost_1 <= cost_2:
                total_harga += list_harga[i]
                f += list_f[i]
                list_accepted[i] = 1
    return list_accepted, f, total_harga

```

Gambar 3.9. Algoritma yang Digunakan  
Sumber: Dokumentasi pribadi

#### IV. KESIMPULAN

Algoritma *branch and bound* dapat digunakan untuk menyelesaikan permasalahan mahasiswa yang harus tetap menjaga kesehatan dan keuangannya sebagai mahasiswa yang sedang merantau menggunakan program yang cukup mudah

dan dapat diimplementasikan menggunakan bahasa Python. Salah satu cabang dari *branch and bound* yang diimplementasikan untuk permasalahan ini adalah *Integer Knapsack Problem*. Program ini dibentuk agar dapat membantu mahasiswa yang sedang merantau untuk mengatasi salah satu permasalahannya.

#### V. PRANALA YOUTUBE

Include link of your video on YouTube in this section.

#### V. UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yang Maha Esa karena atas berkat dan rahmatnya, sehingga penulis dapat menyelesaikan karya tulis dengan tepat waktu, dengan judul “Aplikasi Algoritma Branch and Bound dalam Penentuan Menu pada Rumah Makan sekitar Institut Teknologi Bandung”. Harapannya adalah agar ilmu ini menjadi bermanfaat untuk pembaca. Ucapan terima kasih tak lupa pula kepada dosen pengampu mata kuliah IF2211 Strategi Algoritma kelas 03, Dwi Hendratomo Widiyantoro, atas segala ilmu dan pedoman yang telah diberikan. Tak lupa, ucapan terima kasih juga untuk teman-teman Penulis karena atas dukungannya dan semangatnya, karya tulis ini dapat diselesaikan dengan baik dan yang terakhir, ucapan terima kasih juga kepada diri saya sendiri karena dapat menyelesaikan karya tulis ini dengan sebaik mungkin dan tepat waktu.

#### VI. REFERENSI

- [1] <https://boulevarditb.com/2018/08/24/di-balik-bisnis-ayam-geprek-crisbar/> diakses pada 7 Mei 2021, 22:04
- [2] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Branch-and-Bound-2021-Bagian1.pdf> diakses pada 7 Mei 2021, 01:15
- [3] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Branchand-Bound-2021-Bagian2.pdf> diakses pada 7 Mei 2021, 15:50
- [4] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Branchand-Bound-2021-Bagian4.pdf> diakses pada 7 Mei 2021, 16:12
- [5] <https://mcdonalds.co.id/about> diakses pada 7 Mei 2021, 20:43
- [6] <https://piptools.net/algoritma-bb-branch-and-bound/> diakses pada 7 Mei 2021, 02:53

#### 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, 26 April 2021



Cynthia Rusadi 13519118