

Implementasi Sederhana Algoritma Greedy pada Knapsack Problem dalam Pemilihan Serial Film Untuk Ditonton

Ardhito Saputra 120140003 (*Author*)

Program Studi Teknik Informatika

Jurusan Teknologi Produksi dan Industri

Institut Teknologi Sumatera, Jl. Terusan Ryacudu Lampung

E-mail : ardhito.120140003@student.itera.ac.id

Abstract— Dalam kehidupan sehari-hari, terdapat banyak sekali permasalahan yang dapat muncul yang membuat kita bingung dalam menentukan sesuatu. Bahkan pada hal sederhana seperti memilih beberapa film dari banyak serial film yang kita ingin tonton selama liburan atau akhir pekan, yang terkadang karena banyak nya episode waktu yang dibutuhkan tidak memungkinkan untuk kita tonton semuanya. Untuk membantu kita menyelesaikan permasalahan tersebut disinilah kita dapat menggunakan peran dari algoritma. Untuk kasus memilih optimasi seperti ini, kita dapat mencari solusinya dengan pendekatan persoalan knapsack problem dan dapat dicari solusinya dengan algoritma greedy. Algoritma greedy sendiri merupakan algoritma yang mempunyai prinsip ambil yang terbaik untuk sekarang tanpa memperdulikan apakah itu terbaik untuk kedepannya. Sedangkan knapsack problem adalah sebuah persoalan dalam memilih sesuatu yang mempunyai bobot dan keuntungan untuk dimasukkan ke dalam suatu wadah yang hanya menampung bobot tertentu.

Keywords—Algoritma; Greedy; Knapsack Problem; Film;

I. PENDAHULUAN

Manusia adalah makhluk yang selalu menginginkan kesenangan. Mencari kesenangan menggunakan berbagai cara seperti melakukan hobi, berolahraga, bermain game, dan sebagainya. Namun, karena selalu mencari kesenangan tersebut terkadang membuat kita kehabisan dan malah merepotkan ataupun membuat pusing diri sendiri.

Salah satu contoh yang akan dibahas pada makalah ini adalah mencari kesenangan dengan cara menonton anime. Terkadang, karena terlalu ingin menonton, seseorang akan mengunduh atau membuat list anime yang sudah terlalu banyak. Akibatnya, setelah tersedia waktu untuk menontonnya, orang tersebut akan bingung mau menonton yang mana terlebih dahulu, agar waktunya bisa terpakai dengan optimal. Oleh karena itu, pada makalah ini akan dibahas bagaimana cara mengoptimalkan pemilihan serial anime menggunakan algoritma greedy dan dengan pendekatan ke persoalan 0/1 Knapsack Problem.



Gambar 1. ilustrasi sedang menonton film, sumber <https://cdn2.static.net/kaltim/foto/bank/images/pexels-ilustrasi-nonton-netflix.jpg>

II. TEORI DASAR

A. Algoritma Greedy

Algoritma greedy adalah metode yang paling populer untuk memecahkan persoalan optimasi. Algoritma ini mempunyai prinsip “take what you can get now” yang berarti ambil apa yang dapat kamu ambil sekarang, tanpa memikirkan langkah kedepannya akan jadi seperti apa.

Algoritma greedy merupakan jenis algoritma yang mencari suatu solusi secara bertahap dengan mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu (optimum lokal) tanpa memperhatikan konsekuensi ke depan. Berharap dengan memilih optimum lokal di setiap langkahnya dapat menuju optimum global pada akhir langkah nya. Terdapat dua macam persoalan optimasi yaitu maksimasi dan minimasi. Maksimasi adalah ketika pencarian solusi dengan mengharapkan hasil yang paling banyak sedangkan minimasi mengharapkan hasil yang paling sedikit.

- Himpunan kandidat C Kumpulan semua kemungkinan yang dapat dipilih menjadi solusi
- Himpunan solusi S Kumpulan dari kemungkinankemungkinan yang telah terbukti dapat mengoptimalkan solusi dalam suatu cara.
- Fungsi seleksi Fungsi yang bertanggung jawab untuk melakukan pengecekan dan memastikan tidak ada kandidat yang melewati batas yang telah ditentukan.
- Fungsi objektif Fungsi yang mengoptimalkan nilai solusi yang telah ditemukan sesuai jalan pengoptimasiannya baik maksimal atau minimal.

Secara konklusi, alur dari algoritma greedy sendiri adalah pencarian suatu himpunan bagian S dari himpunan kandidat C dimana S harus memenuhi kriteria yang telah ditetapkan oleh fungsi seleksi yang ada sehingga dapat dinyatakan bahwa himpunan bagian S adalah himpunan solusi yang telah dioptimasi oleh fungsi objektif.

Algoritma greedy sangat sering digunakan untuk menyelesaikan masalah sederhana yang membutuhkan penyelesaian secara cepat dalam computer science. Contoh dari skema pengaplikasian algoritma greedy sendiri dapat dituangkan dalam pseudocode berikut

```
function greedy(input C: himpunan_kandidat) → himpunan_kandidat
{ Mengembalikan solusi dari persoalan optimasi dengan algoritma greedy
Masukan: himpunan_kandidat C
Keluaran: himpunan solusi yang bertipe himpunan_kandidat
}
Deklarasi
x : kandidat
S : himpunan_kandidat
Algoritma:
S ← {} { inisialisasi S dengan kosong }
while (not SOLUSI(S)) and (C ≠ {} ) do
  x ← SELEKSI(C) { pilih sebuah kandidat dari C }
  C ← C - {x} { elemen himpunan_kandidat berkurang satu }
  if LAYAK(S ∪ {x}) then
    S ← S ∪ {x}
  endif
endwhile
{ SOLUSI(S) or C = {} }
if SOLUSI(S) then
  return S
else
  write('tidak ada solusi')
endif
```

Gambar 2. pseudocode dari algoritma greedy secara umum, sumber: https://www.academia.edu/35149365/Algoritma_Greedy_Bahan_Kuliah_IF2211_Strategi_Algoritma

Algoritma greedy sangat sering digunakan untuk menyelesaikan masalah sederhana yang membutuhkan penyelesaian secara cepat dalam computer science. Contoh dari skema pengaplikasian algoritma greedy sendiri dapat dituangkan dalam pseudocode berikut

B. Knapsack Problem

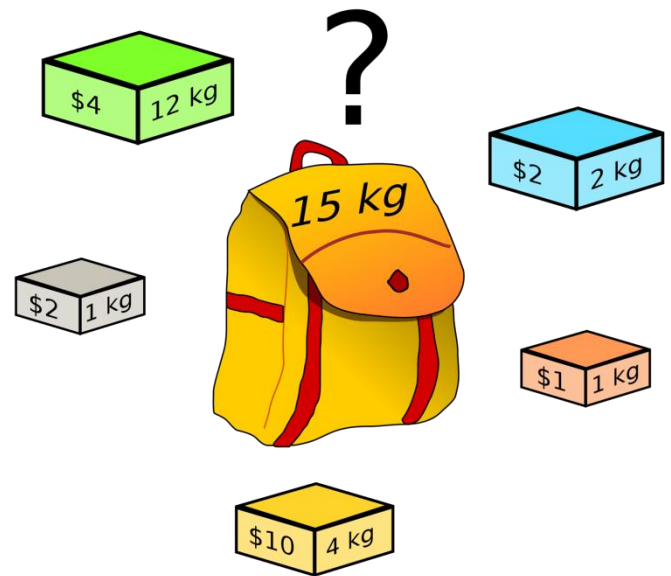
Knapsack problem adalah sebuah permasalahan ketika seseorang berhadapan dengan optimasi pemilihan benda mana yang bisa ditampung ke dalam suatu wadah berkapasitas terbatas. Optimasi ini adalah bagaimana caranya dalam proses pemilihan benda mana yang hendak dimasukkan ke dalam suatu wadah yang dimaksud dihasilkan keuntungan semaksimal mungkin. Masing-masing dari benda tersebut mempunyai bobot nilai keuntungannya telah difungsikan dalam menentukan prioritasnya dalam pemilihan tersebut.

Nilai keuntungan disini bisa berarti harga jual barang, keuntungan penjualan barang, nilai kepentingan, nilai sejarah, dan sebagainya. Wadah disini juga berarti sebuah konstanta pembatas sehingga bobot tidak melewati kapasitas wadah tersebut. Sedangkan bobot disini bisa berarti harga, berat, ukuran, waktu yang dibutuhkan, dan sebagainya untuk benda tersebut.

Terdapat beberapa jenis dari knapsack problem yaitu:

- 0/1 knapsack ketika benda yang tersedia hanya satu, jadi pilihannya hanya diambil atau tidak.
- Fractional knapsack ketika benda pilihannya bisa dipilih sebagian saja, tidak perlu utuh.

- Integer knapsack yaitu masing masing benda mempunyai batas ketersediaan n benda (n bisa tak terbatas) dan hanya bisa dipilih secara utuh.



Gambar 3. ilustrasi knapsack, sumber: <https://www.google.com>

Pada Algoritma Greedy dalam memecahkan persoalan knapsack memiliki tiga strategi dalam memilih objek mana yang hendak dimasukkan ke dalam wadah. Ketiga strategi tersebut antara lain sebagai berikut.

- Greedy by weight

Strategi ini menekankan pada bobot yang paling ringan dalam memasukkan suatu objek. Strategi ini berusaha memaksimalkan keuntungan dari banyaknya objek yang masuk. Dengan kata lain keuntungan akan diperoleh jika objek dimasukkan sebanyak mungkin ke dalam wadah knapsack. Mekanismenya dimulai dengan mengurutkan ke atas objek-objek yang dimaksud dengan berdasarkan pada beratnya. Berikutnya satu demi satu objek diambil hingga tak ada satu pun objek yang dapat dimasukkan atau algoritma knapsack penuh.

- Greedy by profit

Teknik ini menekankan pada objek dengan keuntungan yang besar sebagai pengisi knapsack. Bisa dikatakan bahwa strategi ini berusaha memaksimalkan keuntungan dengan cara mendahulukan yang paling menguntungkan dalam memenuhi space yang tersedia. Prosesnya diawali dengan mengurutkan ke bawah objek-objek dengan berdasarkan pada profitnya. Barulah kemudian satu demi satu objek yang masih dapat ditampung dimasukkan ke dalam knapsack hingga tak ada lagi objek yang dapat dimasukkan atau knapsack penuh.

- Greedy by density

Strategi ini menekankan pada objek-objek dengan densitas terbesar dalam memenuhi wadah yang tersedia. Strategi ini berusaha memaksimalkan keuntungan yang ada dengan cara memilih objek dengan keuntungan tiap unit berat yang paling besar. Mekanismenya diawali dengan mencari nilai keuntungan per unit atau densitas atas masing-

masing objek. Selanjutnya, objek-objek yang dimaksud diurutkan dari segi densitasnya. Barulah satu demi satu dari objek yang masih dapat ditampung dimasukkan hingga penuh ke dalam knapsack.

C. Film

Film adalah gambar-hidup yang juga sering disebut movie. Film secara kolektif sering disebut sebagai sinema. Sinema itu sendiri bersumber dari kata kinematik atau gerak. Film juga sebenarnya merupakan lapisan-lapisan cairan selulosa, biasa di kenal di dunia para sineas sebagai seluloid. Pengertian secara harafiah film (sinema) adalah Cinemathographie yang berasal dari Cinema + tho = phytos (cahaya) + graphie = grhap (tulisan = gambar = citra), jadi pengertiannya adalah melukis gerak dengan cahaya. Agar kita dapat melukis gerak dengan cahaya, kita harus menggunakan alat khusus, yang biasa kita sebut dengan kamera. Film adalah sekedar gambar yang bergerak, adapun pergerakannya disebut sebagai intermitten movement, gerakan yang muncul hanya karena keterbatasan kemampuan mata dan otak manusia menangkap sejumlah pergantian gambar dalam sepersekian detik.

Film menjadi media yang sangat berpengaruh, melebihi mediamedia yang lain, karena secara audio dan visual dia bekerja sama dengan baik dalam membuat penontonnya tidak bosan dan lebih mudah mengingat, karena formatnya yang menarik. 12 Pusat Apresiasi Film Definisi Film Menurut UU 8/1992, adalah karya cipta seni dan budaya yang merupakan media komunikasi massa pandang-dengar yang dibuat berdasarkan asas sinematografi dengan direkam pada pita seluloid, pita video, piringan video, dan/atau bahan hasil penemuan teknologi lainnya dalam segala bentuk, jenis, dan ukuran melalui proses kimiawi, proses elektronik, atau proses lainnya, dengan atau tanpa suara, yang dapat dipertunjukkan dan/atau ditayangkan dengan sistem Proyeksi mekanik, eletronik, dan/atau lainnya.

Film dapat dikategorikan menjadi dua jenis berdasarkan durasinya yaitu the movie dan series. The movie, pada dasarnya sama dengan film lepas pada umumnya yang hanya berjalan satu, dua, atau tiga jam saja. Selama film berlangsung, penonton tidak akan menemukan jeda iklan. Sedangkan pada film series film dapat berepisode-episode dan terdapat jeda waktu dalam penayangan nya biasa nya sekitar satu minggu sekali dikeluarkan episode terbaru nya. Dalam konteks persoalan knapsack ini film yang dimaksud adalah film series yang sudah complete episode nya jadi dapat ditonton dari awal episode sampai akhir episode tanpa jeda.

Untuk menilai bagus tidaknya sebuah film sudah pasti sangat sulit karena penilaian setiap orang berbeda-beda. Karena semua orang pasti punya selera yang berbeda terhadap suatu cerita, penilaian yang berbeda terhadap kualitas animasinya, dan lain sebagainya. Namun di jaman yang sudah modern ini, kita sudah dapat melihat penilaian dari ribuan orang terhadap sebuah film dengan cara melihat rating nya seperti pada web m.imdb.com kita dapat mencari rating dari sebuah film.

Rank & Title	IMDb Rating	Your Rating
1. The Shawshank Redemption (1994)	9.2	
2. The Godfather (1972)	9.1	
3. The Godfather: Part II (1974)	9.0	
4. The Dark Knight (2008)	9.0	
5. 12 Angry Men (1957)	8.9	

Gambar 4. pseudocode dari algoritma greedy secara umum, sumber: <https://medium.com/>

III. IMPLEMENTASI

Menonton film terkadang memakan waktu yang sangat banyak untuk menyelesaikan ceritanya, apalagi jika episodenya sangat banyak. Hal itu terkadang membuat kita harus menunggu waktu luang selanjutnya untuk melanjutkannya dan membuat kita terus penasaran sehingga akhirnya terkadang menjadi gangguan bagi pekerjaan kita sehari-hari. Oleh karena itu, diperlukan kebijakan tertentu dalam memilih serial film untuk ditonton saat liburan sehingga waktu yang tersedia bisa dimanfaatkan dengan maksimal.

A. Aplikasi

Dalam pemilihan film untuk ditonton, ada beberapa hal dari film yang perlu diperhatikan. Setiap film memiliki rating-nya masing-masing dimana seperti yang telah dijelaskan, Akan semakin memuaskan suatu film jika ratingnya semakin tinggi. Jika kita hanya langsung mencoba untuk menonton suatu serial film tanpa melihat ratingnya terlebih dahulu, dikhawatirkan tidak akan suka dan berhenti di tengah-tengah. Jika hal itu terjadi, waktu yang dihabiskan untuk menontonnya akan jadi terbuang sia-sia. Mempertimbangkan hal tersebut, hal ini dapat kita hindari dengan mengaplikasikan strategi algoritma yang ada. Pada kasus ini, akan digunakan algoritma greedy dengan prinsip menyelesaikan sebanyak mungkin film dengan mempertimbangkan ratingnya dengan waktu yang terbatas. Algoritma ini dapat dirancang dengan memasukkan variabelvariabel yang ada kedalam elemen-elemen algoritma greedy. Elemen Algoritma Greedy:

- Himpunan kandidat C: Kumpulan semua film yang dapat ditonton dengan total waktu yang dihabiskan untuk menamatkannya tidak lebih besar dari waktu yang tersedia.
- Himpunan solusi S: Kumpulan film dengan rating tertinggi dari list film yang ingin ditonton dan total waktu yang dihabiskan tidak melebihi waktu yang tersedia
- Fungsi seleksi: Fungsi yang melakukan pengecekan bahwa total waktu yang dihabiskan untuk menamatkan serial film yang ingin ditonton tidak melebihi waktu yang tersedia
- Fungsi objektif: Fungsi yang memastikan bahwa himpunan solusi sudah memenuhi constraint dan berisi

kumpulan film dengan rating terbaik.

B. Studi Kasus

Dalam liburan semester yang berlangsung 1 bulan, penulis ingin menghabiskan waktu untuk menonton film maksimal 2 jam sehari. Penulis memiliki banyak film yang ingin ditonton, namun terbatas waktu sehingga tidak bisa menyelesaikan semuanya. Penulis tidak ingin menonton serial film kemudian berhenti di tengah-tengah karena tidak suka meninggalkan suatu cerita dalam waktu yang lama dan tidak tau kapan lagi punya kesempatan untuk lanjut menontonnya. Penulis bingung harus menonton film yang mana dari daftar film yang ingin ditontonnya sehingga dalam waktu yang tersedia tersebut, penulis ingin menonton sebanyak mungkin serial film dengan rating yang lumayan tinggi sampai selesai. Pilihkanlah film yang sebaiknya ditonton untuk mengoptimalkan waktu yang dipunyai oleh penulis!

Pada kasus yang telah diberikan diatas, kasus tersebut dapat didekati oleh persoalan 1/0 knapsack problem dengan kapasitas knapsack adalah $1 \times 30 \times 2 \text{ jam} = 60 \text{ jam}$. Sehingga total episode yang dapat ditonton adalah $60 \text{ jam} \div 20 \text{ menit/episode} = 180 \text{ episode}$. Berikut adalah daftar serial film beserta jumlah episode dan ratingnya yang ingin ditonton penulis

No	Judul	Rating	Episode
1	Hunter x Hunter (2011)	9.08	144
2	Gintama°	9.05	51
3	Clannad: After Story	8.95	24
4	Shigatsu wa Kimi no Uso	8.71	22
5	Death Note	8.63	37
6	Boku Dake Ga Inai Machi	8.35	12
7	Hyouka	8.13	22
8	Beelzebub	7.90	60
9	Kyoukai no Kanata	7.76	12
10	Mobile Fighter G Gundam	7.56	49

C. Pengujian

Untuk menyelesaikan permasalahan memilih serial film untuk ditonton ini, dapat didekatkan dengan metode penyelesaian persoalan knapsack problem menggunakan algoritma greedy, yaitu dengan greedy by weight, greedy by profit, greedy by density dalam hal ini, weight adalah jumlah episode, value adalah rating, dan density adalah rating/episode. Pertama, lihat dulu himpunan kandidatnya. Perhatikan bahwa, film Naruto melebihi jumlah episode yang tidak dapat ditonton saat liburan, sehingga Naruto bukanlah himpunan kandidat yang dapat dipilih.

1) Greedy by Episode

Pada metode ini, kita memaksimalkan banyaknya film yang ditonton dengan memilih film menggunakan langkah-langkahnya adalah sebagai berikut:

- Pada setiap Langkah, pilih film dengan jumlah

episode paling kecil. (akan lebih mudah jika di awal sudah diurutkan terlebih dahulu)

- Jika ada yang sama, maka ambil yang ratingnya paling tinggi terlebih dahulu
- Cek apabila yang akan ditonton waktunya masih cukup menyelesaikannya.
- Jika cukup maka ditonton, jika tidak maka ditinggalkan. Sehingga kita akan mendapatkan film yang akan ditonton adalah sebagai berikut:

No	Judul	Rating	Episode	Dipilih
1	Boku Dake Ga Inai Machi	8.35	12	1
2	Kyoukai no Kanata	7.76	12	1
3	Shigatsu wa Kimi no Uso	8.71	22	1
4	Hyouka	8.13	22	1
5	Clannad: After Story	8.95	24	1
6	Death Note	8.63	37	1
7	Mobile Fighter G Gundam	7.56	49	1
8	Beelzebub	7.90	60	0
9	Gintama°	9.05	51	0
10	Hunter x Hunter	9.08	144	0
Total rating: 58.09				
Total episode: 178				

2) Greedy by Rating

Pada metode ini, kita memaksimalkan tingkat kepuasan dengan memilih film dengan rating yang paling tinggi terlebih dahulu, sehingga prosesnya adalah sebagai berikut:

- Pada setiap Langkah, pilih film dengan rating paling tinggi.
- Apabila ada rating yang sama, maka pilih yang episodenya paling sedikit terlebih dahulu.
- Cek apabila waktu yang tersedia masih cukup atau tidak untuk menyelesaikannya
- Jika waktu masih cukup, maka dipilih untuk ditonton, jika tidak maka tidak dipilih dan maju ke Langkah selanjutnya.

Sehingga solusinya adalah :

No	Judul	Rating	Episode	Dipilih
1	Hunter x Hunter (2011)	9.08	144	1
2	Gintama°	9.05	51	0
3	Clannad: After Story	8.95	24	1
4	Shigatsu wa Kimi no Uso	8.71	22	0
5	Death Note	8.63	37	0
6	Boku Dake Ga Inai Machi	8.35	12	1
7	Hyouka	8.13	22	0
8	Beelzebub	7.90	60	0

9	Kyoukai no Kanata	7.76	12	0
10	Mobile Fighter G Gundam	7.56	49	0
Total rating: 26.38				
Total episode: 180				

3) Greedy by Density

Pada metode ini, kita memaksimalkan densitinya yaitu rating dibagi dengan jumlah episode, sehingga prosesnya adalah sebagai berikut

- Hitung density dari semua himpunan kandidat.
- Pada setiap Langkah, pilih film dengan density paling tinggi.
- Apabila ada rating yang sama, maka pilih yang episodenya paling sedikit terlebih dahulu
- Cek apabila waktu yang tersedia masih cukup atau tidak untuk menyelesaikannya
- Jika cukup maka ditonton, jika tidak maka ditinggalkan.

Sehingga solusinya :

No	Judul	Density	Episode	pilih
1	Boku Dake Ga Inai Machi	0.69	12	1
2	Kyoukai no Kanata	0.64	12	1
3	Shigatsu wa Kimi no Uso	0.39	22	1
4	Hyouka	0.37	22	1
5	Clannad: After Story	0.372	24	1
6	Death Note	0.23	37	1
7	Gintama°	0.18	51	1
8	Mobile Fighter G Gundam	7.56	49	0
9	Beelzebub	7.90	60	0
10	Hunter x Hunter	0.06	144	0
Total rating: 59.58				
Total episode: 180				

Bisa kita lihat bahwa greedy menggunakan acuan rating hanya menghasilkan 3 jenis film yaitu HunterxHunter, Clannad AfterStory, dan Boku dake ga inai machi, sedangkan greedy menggunakan acuan densitas ataupun jumlah episode mendapatkan 7 jenis film walaupun terdapat satu perbedaan. Dari total rating dan total episodenya, greedy by density masih lebih baik dalam memanfaatkan waktu yang tersedia dengan total rating yang paling tinggi dibanding yang lainnya.

IV. KESIMPULAN

Dalam makalah ini, penulis telah membahas tentang penerapan algoritma greedy dalam salah satu bidang kehidupan manusia sehari-hari yaitu dalam pemilihan film untuk ditonton. Berdasarkan percobaan yang dilakukan, Algoritma greedy ini terutama yang menggunakan acuan densitas sudah menghasilkan hasil yang optimal dalam memilih anime untuk ditonton.

Makalah ini hanyalah salah satu contoh saja dari penerapan algoritma greedy, tidak hanya untuk film, algoritma ini juga masih bisa dilanjutkan untuk permasalahan permasalahan optimasi yang lainnya.

Tentu saja, algoritma greedy bukan satu-satunya yang dapat digunakan untuk menyelesaikan masalah seperti ini, namun masih banyak algoritma lain yang lebih efektif dan efisien. Hanya saja, penulis berusaha menunjukkan bahwa Algoritma greedy untuk menyelesaikan masalah ini terbilang cukup sederhana dan mudah dimengerti.

VI. REFERENSI

- [1] [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) diakses pada 11 mei 2021 pukul 14.00 WITA.
- [2] [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) diakses pada 11 Mei 2021 pukul 14.00 WITA.
- [3] Lesley Aeschliman, [What Is Anime? - Anime \(bellaonline.com\)](http://bellaonline.com) diakses pada 11 Mei 2021 pukul 14.00 WITA.
- [4] Marzuki, [Algoritma Knapsack Problem - DosenIT.com](http://DosenIT.com), diakses pada 11 Mei 2021 pukul 14.00 WITA.

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.

Lampung, 29 Maret 2022



Ardhito Saputra, 120140003

