

Implementasi Algoritma *Greedy* pada Permainan *Hangman* Versi Bahasa Indonesia dengan Memanfaatkan Frekuensi Huruf

Diki Ardian Wirasandi - 13515092

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13515092@std.stei.itb.ac.id

Abstrak—Permasalahan pada permainan *Hangman* dapat diselesaikan dengan memasukkan huruf-huruf penyusun kata yang harus ditebak. Dalam pemilihan huruf-huruf tersebut tidak bisa dengan asal pilih saja karena pemain diberikan batas maksimal kesalahan. Untuk itu diperlukan strategi khusus dalam memilih huruf-huruf penyusun kata tersebut, salah satunya yaitu dengan menerapkan algoritma *greedy*.

Kata kunci—*Hangman*, algoritma, *greedy*, huruf, kata, frekuensi.

I. PENDAHULUAN

Dalam kehidupan sehari-hari, kita sering dihadapkan kepada beberapa persoalan yang harus kita selesaikan tentunya. Apalagi dalam bermain *games* atau permainan, pastinya pemain akan dihadapkan suatu persoalan atau tantangan dan mau tidak mau pemain harus bisa menyelesaikan tantangan tersebut agar memperoleh nilai atau skor.

Pada permainan dengan tipe strategi, pemain harus bisa menyelesaikan permasalahan yang diberikan menggunakan siasat-siasat dan dibutuhkan pemikiran serta pertimbangan-pertimbangan yang harus diperhatikan. Salah satu permainan yang terkenal dengan siasat untuk menyelesaikannya adalah permainan *Hangman*. Dalam permainan ini, pemain diminta dapat menebak kata (satu atau lebih) dan yang diketahui hanya panjang katanya saja (jumlah huruf). Namun, untuk memudahkan pemain agar cakupan kata-katanya menjadi lebih sempit, biasanya diberikan bantuan misalkan kata tersebut masuk ke dalam kategori apa, dan sebagainya.

Pemain diminta memasukkan huruf per huruf yang mungkin menyusun kata tersebut. Namun pemain tidak bisa asal memasukkan huruf begitu saja. Karena setiap kesempatan menebak kata, pemain diberikan nilai maksimal kesalahan. Jika pemain melakukan kesalahan melebihi nilai maksimal yang diberikan, maka pemain gagal dalam menebak kata tersebut.

Karena adanya nilai maksimal kesalahan tersebut, maka dibutuhkan strategi untuk menebak kata yang diberikan. Salah satu strategi yang dapat diterapkan yaitu strategi algoritma *greedy*. Dengan menerapkan strategi ini, diharapkan pemain

dapat menebak kata yang diajukan tanpa melakukan kesalahan melebihi nilai maksimal yang ditentukan.

Makalah ini akan membahas strategi algoritma *greedy* yang digunakan dalam menyelesaikan permasalahan pada permainan *Hangman* versi bahasa Indonesia dengan memanfaatkan frekuensi kemunculan huruf pada kata-kata.

II. LANDASAN TEORI

A. Algoritma *Greedy*

Algoritma *Greedy* merupakan jenis algoritma penyelesaian masalah dengan menggunakan pendekatan mencari nilai maksimum sementara pada setiap langkahnya [3]. Algoritma *Greedy* membentuk solusi langkah per langkah (*step by step*) [1]. Pada dasarnya, pada setiap langkah solusi akan terdapat banyak pilihan yang perlu dilakukan eksplorasi. Oleh karena itu, pada setiap langkahnya akan dibuat suatu keputusan yang terbaik (sementara) dalam menentukan pilihan yang kemudian berlanjut ke langkah selanjutnya hingga menemukan solusi atau tujuan (*goal*).

Sebagai contoh penerapannya dalam masalah klasik yaitu mencari sirkuit atau lintasan terpendek dari titik A ke titik B, yaitu dengan dari titik A memilih sirkuit terpendek (sementara) pada setiap langkahnya, begitu seterusnya hingga mencapai titik B. Dengan kata lain *algoritma greedy* [1]:

1. Memilih pilihan terbaik yang dapat diperoleh saat itu (optimum lokal) tanpa memperhatikan apa yang terjadi pada langkah berikutnya.
2. Diharapkan dengan memilih optimum local pada setiap langkahnya dapat berakhir dengan optimum global, yaitu optimum untuk solusi yang dituju.

Pada pencarian solusi yang optimal dalam konteks algoritma *greedy*, disusun oleh elemen-elemen sebagai berikut:

- Himpunan kandidat
Himpunan yang berisi elemen-elemen pembentuk solusi

- Himpunan solusi
Himpunan yang berisi kandidat-kandidat yang terpilih sebagai solusi persoalan.
- Fungsi seleksi
Fungsi yang pada setiap langkah memilih kandidat yang paling memungkinkan mencapai solusi optimal.
- Fungsi kelayakan
Fungsi Fungsi yang memeriksa apakah suatu kandidat yang dipilih dapat memberikan solusi yang layak atau dengan kata lain tidak melanggar batasan (*constraint*).
- Fungsi obyektif
Fungsi yang mengoptimalkan nilai solusi.

Skema umum yang digunakan pada algoritma *greedy* dapat dirumuskan sebagai berikut:

1. Inisialisasi himpunan solusi dengan kosong;
2. Pilih sebuah kandidat dengan memanfaatkan fungsi seleksi dari himpunan kandidat;
3. Kurangi himpunan kandidat dengan kandidat yang sudah terpilih pada langkah 2.
4. Lakukan pemeriksaan apakah kandidat yang dipilih tersebut bersama dengan himpunan solusi membentuk solusi yang layak;
5. Jika layak, masukkan ke dalam himpunan solusi, jika tidak layak, buang kandidat tersebut.
6. Periksa dengan fungsi solusi apakah himpunan solusi sudah mencapai solusi yang lengkap, jika belum maka ulangi dari langkah 2.

Perlu ditekankan bahwa dalam pemilihan keputusan dari setiap langkah (optimum lokal), jika keputusan sudah diambil maka tidak dapat diulangi ke tahap sebelumnya kembali, dan pada penerapannya, algoritma *greedy* tidak dapat memprediksi langkah selanjutnya, sehingga hanya memperhitungkan langkah pada saat itu saja.

Algoritma *Greedy* ini banyak diterapkan pada beberapa contoh permasalahan, seperti masalah penukaran uang, minimisasi penjadwalan, persoalan *knapsack*, permasalahan pohon merentang minimum, permasalahan lintasan terpendek, TSP dan lain-lain.

B. Permainan Hangman

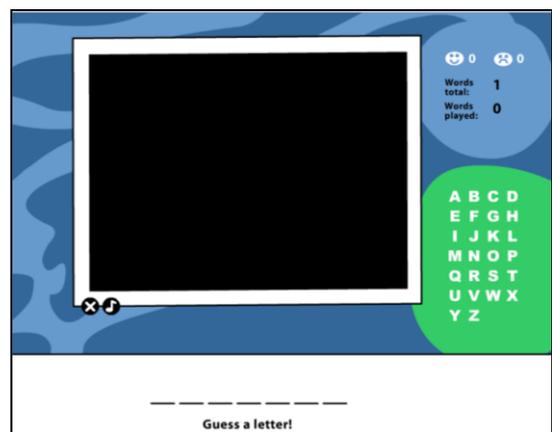
Hangman adalah sebuah permainan dengan tipe permainan *puzzle*. *Hangman* dapat dimainkan oleh dua orang atau lebih. Orang pertama berperan sebagai “pembuat kata” rahasia, sedangkan orang yang lainnya berperan sebagai “penebak kata” rahasia tersebut [4]. Dalam setiap percobaan menebak kata, pemain penebak diberi kesempatan maksimal kesalahan huruf. Jadi, penebak kata harus menebak huruf per huruf yang

menyusun kata misteri tersebut. Penebak hanya mengetahui panjang katanya saja.

Kata *hangman* berasal dari bahasa Inggris yang berarti orang yang “bergantung”. Ini diambil dari cara menghitung kesalahan huruf yang diajukan oleh penebak kata. Setiap kesalahan akan ditandai dengan satu goresan pada kertas, yang nantinya goresan-goresan tersebut akan membentuk gambar orang “gantungan diri”. Biasanya jumlah maksimal kesalahan (jumlah goresan) adalah 6 sampai 10.

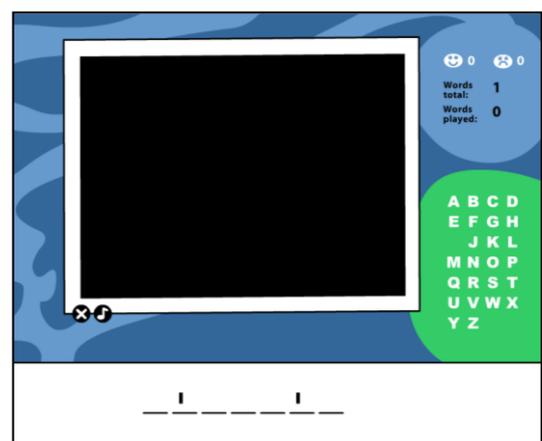
Dalam perkembangannya, permainan *hangman* ini dapat dimainkan oleh satu orang saja dengan memainkannya dalam bentuk *game* pada komputer atau *smartphone*, contohnya pada halaman *website* <http://www.hangman.no/>. Kata misterius akan otomatis telah disediakan dan pemain hanya sebagai penebak kata. Penambahan petunjuk juga dapat memudahkan dalam menebak kata, seperti kategori benda (makanan, hewan, nama kota, dll.) dan sebagainya. Sebagai contoh, diberikan kata misterius “BISKUIT”, maka penebak harus memasukkan huruf satu per satu hingga kata biskuit terbentuk sempurna, misalkan ditambahkan petunjuk yaitu kategori makanan.

Gambar 2.1 Hangman dengan kata “BISKUIT”, sumber:



<http://www.hangman.no/>

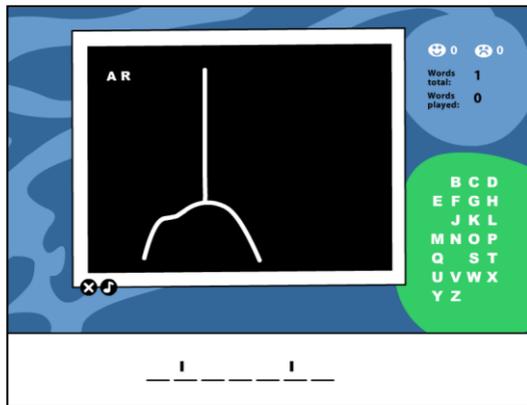
Terlihat ada kata dengan panjang 7 huruf yang harus ditebak. Misalkan penebak memasukkan huruf “I”.



Gambar 2.2 Hangman dengan kata “BISKUIT”, sumber:

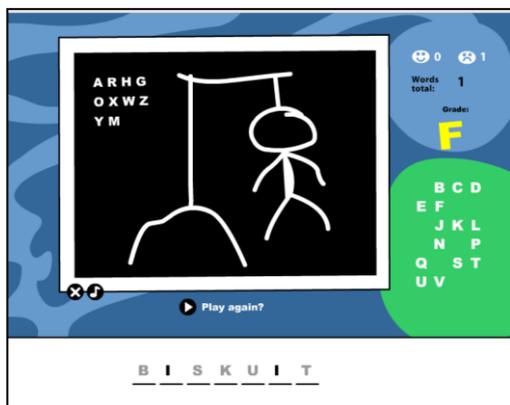
<http://www.hangman.no/>

Maka akan terbuka semua letak huruf ‘I’ pada kata misterius tersebut. Misalkan dimasukkan lagi huruf ‘A’ kemudian huruf ‘R’,



Gambar 2.3 Hangman dengan kata ‘BISKUIT’, sumber: <http://www.hangman.no/>

Tidak ada huruf yang terbuka karena ‘A’ dan ‘R’ bukan huruf penyusun kata misterius ‘BISKUIT’. Terlihat juga ada dua goresan pada permainan yang menandakan jumlah kesalahan huruf. Jika kesalahan mencapai maksimal (pada permainan ini 8 kali kesalahan), akan terbentuk gambar *hangman* atau orang ‘gantungan diri’ seperti pada Gambar 2.4 dan akan ditampilkan kata misterius yang seharusnya ditebak.



Gambar 2.4 Hangman dengan kata ‘BISKUIT’, sumber: <http://www.hangman.no/>

Jika penebak kata dapat membuka semua huruf penyusun kata misterius dengan kesalahan kurang dari batas maksimal yang ditentukan (dalam permainan ini maksimal 8), maka si penebak berhasil menebaknya dan tidak terbentuk gambar *hangman* dengan sempurna.

Kata misterius yang diajukan dapat lebih dari satu kata (dua atau tiga) yang dapat menambah tingkat kesulitan bagi pemain yang berperan sebagai penebak kata. Permainan *Hangman* ini, selain menghibur, juga dapat mengasah otak dan dapat juga untuk melatih anak dalam mengeja kata-kata yang baru dipelajarinya.

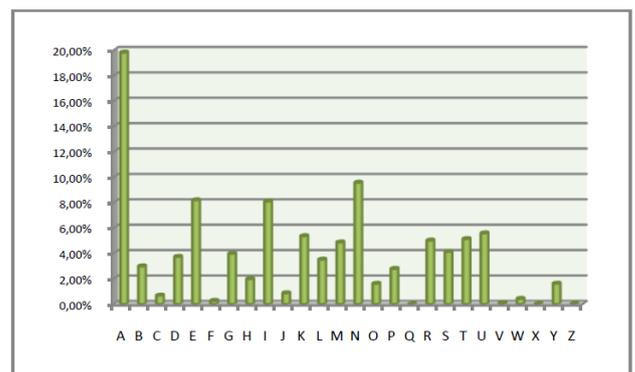
III. PEMBAHASAN

A. Frekuensi Huruf

Setiap bahasa di dunia ini pasti memiliki kumpulan kata-kata yang menyusun bahasa tersebut. Setiap kata-kata tersusun atas beberapa huruf, baik itu huruf vokal maupun konsonan. Huruf-huruf penyusun kata dalam bahasa tersebut dapat dihitung kemunculannya sehingga dapat diketahui mana huruf yang sering dipakai dan mana yang jarang sekali dipakai. Kemunculan huruf tersebut dapat disebut dengan istilah frekuensi huruf (*letter frequency*).

Dalam bahasa Inggris misalnya, frekuensi huruf dapat ditampilkan dalam Grafik 1.1 *English Letter Frequency*, sumber: [5] pada bagian Lampiran. Dari grafik tersebut dapat dilihat bahwa lima huruf yang sering muncul dalam bahasa Inggris adalah E, T, A, O dan I.

Dari bahasa Indonesia tentunya dapat dihitung frekuensi huruf, melalui beberapa percobaan dari lebih dari 26.000 kata[2]. Didapatkan hasilnya, bahwa frekuensi huruf bahasa Indonesia sebagai berikut:



Grafik 3.1 Frekuensi huruf dalam bahasa Indonesia, sumber: [2]

Berdasarkan data yang diperoleh [2] seperti pada grafik di atas, urutan huruf-huruf yang kemunculannya dari yang lebih sering menuju ke yang lebih jarang adalah:

A, N, E, I, U, K, T, R, M, S, G, D, L, B, P, H, O, Y, J, C, W, F, V, Z, X, Q

Dari data tersebut akan dimanfaatkan dalam penyusunan strategi algoritma *greedy* dalam menebak huruf-huruf penyusun kata misterius pada permainan *Hangman*.

B. Implementasi Algoritma Greedy pada Permainan Hangman

Fokus dari permainan *Hangman* yang dibahas pada makalah ini adalah permainan *Hangman* versi bahasa Indonesia, artinya kata misterius yang digunakan adalah kosakata dalam bahasa Indonesia, seperti contoh pada bagian II, yaitu ‘BISKUIT’. Strategi algoritma *greedy* yang akan digunakan memanfaatkan data frekuensi huruf seperti yang telah dijelaskan di atas, tentunya frekuensi huruf bahasa Indonesia. Alasan pemanfaatan frekuensi huruf pada strategi *greedy* ini adalah dengan mempertimbangkan peluang benar pada penebakan huruf kata misterius akan lebih tinggi dengan urutan menebak sesuai dengan frekuensi huruf. Huruf dengan

frekuensi kemunculan yang tinggi akan besar kemungkinannya dimiliki oleh suatu kata, begitu pun pada kata misterius yang akan dipecahkan ini. Namun, ada beberapa sedikit modifikasi dan optimalisasi kemungkinan yang akan diterapkan lebih lanjut. Jadi, tidak hanya mempertimbangkan urutan yang sesuai persis dengan data frekuensi huruf. Beberapa kemungkinan, kemungkinan yang dipertimbangkan adalah:

- Huruf vokal yang lebih didahulukan daripada huruf konsonan, karena setiap kata dalam bahasa Indonesia pasti memiliki huruf vokal.
- Huruf “G” dan “Y” yang diprioritaskan setelah terbukti bahwa huruf “N” terdapat dalam kata misterius, karena kombinasi dua huruf ini (“NG” dan “NY”) sering digunakan didalam bahasa Indonesia.
- Berdasarkan data frekuensi huruf, akan dibagi menjadi dua golongan, yaitu huruf vokal dan huruf konsonan, sehingga urutan huruf vokal (sesuai frekuensi huruf dari tinggi ke rendah) adalah:

A, E, I, U, O

Sedangkan, urutan huruf konsonan (sesuai frekuensi huruf dari tinggi ke rendah) adalah:

N, K, T, R, M, S, G, D, L, B, P, H, Y, J, C, W, F, V, Z, X, Q

Dari golongan huruf konsonan, akan dibagi lagi menjadi tiga golongan, yaitu golongan sering muncul, yaitu:

Golongan 1 (Sering muncul; n = 2)	Golongan 2 (Frekuensi muncul menengah; n = 2)	Golongan 3 (Jarang muncul; n = 1)
N, K, T	R, M, S, G, D	L, B, P, H, Y, J

Tabel 3.1 Penggolongan huruf konsonan

Terlihat bahwa huruf konsonan C, W, F, V, Z, X dan Q tidak masuk golongan manapun, karena memiliki frekuensi kemunculan yang terlalu rendah sehingga kemunculannya sangat jarang, sehingga disini dapat diabaikan. Penggolongan ini didasarkan atas asumsi bahwa suatu kata, rata-rata terdiri atas huruf-huruf yang sering muncul (golongan 1 dan 2) dan juga yang jarang muncul dengan porsi yang lebih sedikit dari yang sering muncul, contohnya kata “BERANGKAT”, terdapat huruf konsonan golongan 1 sebanyak 3 huruf, golongan 2 sebanyak 2 huruf dan golongan 3 sebanyak 1 huruf. Dengan pertimbangan seperti ini akan meningkatkan peluang menebak huruf-huruf pada kata misterius permainan *Hangman*. Nilai n yang dimiliki di setiap golongan adalah nilai prioritas yang akan dijelaskan berikutnya.

Perlu diperhatikan bahwa tujuan utama penerapan algoritma greedy pada permainan *Hangman* disini adalah untuk membantu mengefisienkan penebakan huruf-huruf sehingga kesalahan yang terjadi dapat seminimal mungkin dan dapat menebak maksimal menyisakan 2 (dua) huruf yang belum tertebak. Jadi, tujuan utamanya bukan menyelesaikan atau menebak seluruh huruf yang menyusun kata misterius yang diajukan, namun tidak menutup kemungkinan untuk menebak huruf-huruf pada suatu kata secara penuh pada kata-kata yang kebetulan dapat diselesaikan. Tentunya, dengan mencapai tujuan ini akan sangat memudahkan pemain dalam menebak kata misterius, misalkan diajukan kata misterius dengan panjang 8 huruf, dengan menerapkan strategi ini, akan tertebak sekurang-kurangnya 6 huruf sehingga akan menyisakan (maksimal) 2 huruf yang harus ditebak (dan seharusnya dapat ditebak) berdasarkan pengetahuan apalagi dibantu oleh kategori kata atau petunjuk yang diberikan dengan jumlah kesalahan yang tersisa yang jangan sampai melampaui batas maksimum kesalahan.

Elemen – elemen yang menyusun strategi ini adalah sebagai berikut:

- Himpunan kandidat
Himpunan semua huruf vokal, dan huruf konsonan golongan 1, 2 dan 3, kemungkinan dapat berubah-ubah tiap tahapnya disesuaikan dengan prioritas untuk optimalisasi.
- Himpunan solusi
Himpunan huruf-huruf yang telah dipilih dari kandidat dan telah diajukan untuk menebak huruf-huruf pada kata misterius, baik itu benar (terandung dalam kata misterius) maupun salah (tidak terandung dalam kata misterius).
- Fungsi seleksi
Fungsi untuk memilih solusi dari kandidat-kandidat disesuaikan dengan kondisi dari kata misterius (berapa huruf yang sudah tertebak, dan berapa yang belum). Fungsi seleksi akan dijelaskan lebih detil berikutnya.
- Fungsi kelayakan
Apakah kandidat yang terpilih sudah berada di himpunan solusi atau belum. Jika sudah ada maka tidak layak, jika belum ada maka layak.
- Fungsi obyektif
Menebak huruf-huruf pada kata misterius dengan menyisakan maksimal 2 huruf yang belum tertebak pada kata tersebut.

Langkah-langkah secara keseluruhan yang diterapkan pada strategi algoritma greedy pada pemilihan huruf untuk menebak kata misteri dalam permainan *Hangman* sebagai berikut:

1. Inisialisasi Himpunan Solusi dengan himpunan kosong.
2. Pilih dari huruf vokal (A, E, I, U, O) dengan pemilihan sesuai urutan satu per satu hingga sukses menebak 2 huruf vokal pertama **atau** telah tertebak minimal setengah huruf dari panjang kata misterius.

Masukkan setiap huruf yang dipilih (sukses maupun gagal) ke himpunan solusi. Jika huruf yang belum tertebak lebih dari 2 huruf **atau** pemain belum bisa menebak sisa huruf yang belum tertebak (lebih dari 2), maka lanjut ke langkah 2. Jika sudah, maka berhenti.

3. Instruksi umum:

Pilih dari huruf konsonan golongan 1 (N, K, T) dengan pemilihan sesuai urutan satu per satu hingga sukses menebak 1 huruf **atau** telah gagal menebak 2 huruf (sesuai nilai $n=2$).

Instruksi khusus:

Kasus khusus jika terpilih “N” dan sukses serta pada kata misterius huruf tepat setelah “N” ada huruf yang belum tertebak, maka selanjutnya adalah pilih (G, Y) sesuai urutan satu per satu hingga sukses menebak 1 huruf **atau** telah gagal menebak dua-duanya (G, Y).

Masukkan setiap huruf yang dipilih (sukses maupun gagal) ke himpunan solusi. Jika huruf yang belum tertebak lebih dari 2 huruf **atau** pemain belum bisa menebak sisa huruf yang belum tertebak (lebih dari 2), maka lanjut ke langkah 3. Jika sudah, maka berhenti.

4. Instruksi umum:

Pilih dari huruf konsonan golongan 2 (R, M, S, G, D) dengan pemilihan sesuai urutan satu per satu hingga sukses menebak 1 huruf **atau** telah gagal menebak 2 huruf (sesuai nilai $n=2$).

Instruksi khusus:

Jika pada langkah 2 terpilih “N” dan sukses, kemudian dipilih “G” dan **sukses** maka langkah 3 ini dapat **dilewati**.

Jika pada langkah 2 terpilih “N” dan sukses, kemudian dipilih “G” dan **gagal** maka sama seperti instruksi umum tetapi **hingga sukses menebak 1 huruf atau telah gagal menebak 1 huruf**.

Masukkan setiap huruf yang dipilih (sukses maupun gagal) ke himpunan solusi. Jika huruf yang belum tertebak lebih dari 2 huruf **atau** pemain belum bisa menebak sisa huruf yang belum tertebak (lebih dari

2), maka lanjut ke langkah 4. Jika sudah, maka berhenti.

5. Instruksi umum:

Pilih dari huruf konsonan golongan 3 (L, B, P, H, Y, J) dengan pemilihan sesuai urutan satu per satu hingga sukses menebak 1 huruf **atau** telah gagal menebak 1 huruf (sesuai nilai $n=1$).

Instruksi khusus:

Jika pada langkah 2 telah dipilih “Y” (baik sukses maupun gagal), maka langkah 4 ini dapat **dilewati**.

Masukkan setiap huruf yang dipilih (sukses maupun gagal) ke himpunan solusi. Jika huruf yang belum tertebak lebih dari 2 huruf **atau** pemain belum bisa menebak sisa huruf yang belum tertebak (lebih dari 2), maka lanjut ke langkah 3. Jika sudah, maka berhenti.

6. Jika huruf yang belum tertebak lebih dari 2 huruf **atau** pemain belum bisa menebak sisa huruf yang belum tertebak (lebih dari 2), maka ulangi kembali dari langkah 2 hingga 5 berlaku untuk huruf-huruf yang belum terpilih (belum ada di himpunan solusi, jika sudah dipilih semua maka dilewati) dan khusus untuk langkah 2 diulangi hanya jika **masih dimungkinkan (perkiraan) untuk ditebaknya huruf vokal**, yaitu dicirikan dengan:

Terdapat dua atau lebih huruf yang berjajar dan belum tertebak dan terdapat huruf konsonan minimal di salah satu sisinya, contoh:

L _ _ _ I A	masih mungkin,
_ _ L A S E	masih mungkin,
A _ _ I	tidak dimungkinkan (perkiraan)

Agar lebih jelas diberikan contoh sebagai berikut:

Kata misteri: _ _ _ _ _ (jawaban: “MONYET”)

Petunjuk/Kategori: BINATANG

Langkah 1:

Kata Misteri	Pilihan Huruf	Himpunan Solusi	Kesalahan (maks. 9)	Ket.
-----	A	{A}	1	Menebak 1 huruf, 1 gagal, pilih lagi
----E--	E	{A, E}	1	1 gagal, 1 sukses, pilih lagi
----E--	I	{A, E, I}	2	1 gagal, 1 sukses, pilih lagi
----E--	U	{A, E, I, U}	3	1 gagal, 1 sukses, pilih lagi
_O__E_	U	{A, E, I, U, O}	3	2 sukses,

Keterangan: Pada kolom himpunan solusi, huruf dengan warna merah berarti gagal menebak, sedangkan warna hitam berarti sukses menebak

Masih belum menyisakan 2 huruf belum tertebak atau pemain belum bisa menebak, lanjut ke langkah berikutnya.

Langkah 2:

Kata Misteri	Pilihan Huruf	Himpunan Solusi	Kesalahan (maks. 9)	Ket.
_ON_E_	U	{A, E, I, U, O, N}	3	1 sukses, karena "N" sukses, maka selanjutnya dipilih dari (G, Y)
_ON_E_	U	{A, E, I, U, O, N, G}	4	"G" gagal, kemudian periksa "Y"
ONYE	U	{A, E, I, U, O, N, G, Y}	4	"Y" sukses,

Keterangan: Pada kolom himpunan solusi, huruf dengan warna merah berarti gagal menebak, sedangkan warna hitam berarti sukses menebak

Tersisa dua huruf yang belum tertebak, maka sudah mencapai *goal*. Dengan bantuan petunjuk dan wawasan yang dimilikinya, maka pemain akan menebak bahwa kata tersebut

adalah "MONYET" dengan memilih "M" dan "T". Didapatkan 4 kesalahan yang artinya masih terdapat 4 kesalahan lagi untuk menebak 2 huruf yang belum tertebak. Terlihat bahwa penggunaan strategi ini sangat membantu menyelesaikan kata misteri yang diberikan.

C. Hasil Percobaan

Untuk membuktikan apakah algoritma ini menghasilkan solusi yang memudahkan dan membantu dalam menebak kata misteri pada permainan *Hangman*, dilakukan beberapa percobaan dengan menggunakan beberapa kata misteri dengan beberapa petunjuk atau kategori kata yang diberikan dengan jumlah kesalahan maksimal 8. Dan hasilnya disajikan pada Tabel 3.2 berikut.

Kategori Kata / petunjuk	Kata misteri dengan huruf tertebak (jawaban)	Kesalahan (maks. 9)	Himpunan Solusi
Alat transportasi	_E_A K (BECAK)	1	{A, E, N, K}
	U K O T A (BUS KOTA)	6	{A, E, I, U, N, K, R, M, L, O, T}
	E M A N (DELMAN)	1	{A, E, N, R, M}
	K E R E _ A A _ I (KERETA API)	2	{A, E, N, K, R, L, I}
	M O B I L B A L A _ (MOBIL BALAP)	8	{A, E, I, N, K, R, M, L, U, O, T, S, G, B}
	_ E S A _ A T T E R B A N G (PESAWAT TERBANG)	3	{A, E, N, G, L, K, T, R, B, M, S}
Buah-buahan	_ I _ A N G (PISANG)	1	{A, E, I, N, G}
	_ E _ U K (JERUK)	3	{A, E, I, U, N, K}
	A L _ U K A _ (ALPUKAT)	5	{A, E, I, U, N, K, R, M, L}
	_ _ R I A N (DURIAN)	1	{A, E, I, N, R}
	_ E M A N G _ A (SEMANGKA)	1	{A, E, N, G, R, M}

	_ A M _ U (JAMBU)	5	{A, E, I, U, N, K, R, M}
	R A _ _ U T A N (RAMBUTAN)	4	{A, E, I, U, N, R, L, K, T}
Alat Sekolah	_ O L _ O I N (BOLPOIN)	5	{A, E, I, U, O, N, R, M, L}
	_ E N S I L _ A R N A (PENSIL WARNA)	5	{A, E, N, G, Y, R, L, I, K, T, M, S}
	_ U K U T U L I _ (BUKU TULIS)	5	{A, E, I, U, N, K, R, M, L, T}
	I K A _ _ I N G G A N G (IKAT PINGGANG)	2	{A, E, I, N, G, L, K}
	_ E R A _ A M (SERAGAM)	4	{A, E, N, K, R, L, T, M}
	B U K U _ A M B A R (BUKU GAMBAR)	5	{A, E, I, U, N, K, R, L, T, M}
	_ E N G G A R I _ (PENGGARIS)	3	{A, E, N, G, L, I, K, T, R}

Keterangan: Pada kolom himpunan solusi, huruf dengan warna merah berarti gagal menebak, sedangkan warna hitam berarti sukses menebak

Tabel 3.2 Hasil Percobaan

Dari hasil percobaan yang dilakukan pada 20 kosakata, tingkat pencapaian tujuan, yaitu menyisakan maksimal 2 huruf yang belum terpecah, adalah 100% atau semua kata dapat mencapai tujuan. Keberagaman huruf dalam kata misteri akan menambah tingkat kesulitan dalam penyelesaiannya sehingga akan menambah jumlah kegagalan. Dengan dilakukannya percobaan di atas, maka lebih membuktikan bahwa strategi algoritma dalam menyelesaikan permainan *Hangman* versi bahasa Indonesia ini dapat membantu pemain untuk menebak kata misteri.

IV. SIMPULAN

Dalam menyelesaikan permainan *Hangman* versi bahasa Indonesia, yaitu menebak kata misteri yang diberikan, dapat dilakukan dengan menggunakan strategi algoritma *greedy* dengan memandaatkan data frekuensi huruf dalam bahasa Indonesia. Hal ini tidak menutup kemungkinan diterapkan pada permainan *Hangman* untuk versi bahasa lain. Tentunya setiap bahasa jika dianalisa memiliki frekuensi huruf masing-masing dan data tersebut dapat dimanfaatkan untuk memudahkan penyelesaian permainan *Hangman* dengan menambahkan prioritas-prioritas tertentu untuk tujuan optimalisasi.

V. UCAPAN TERIMA KASIH

Penulis mengucapkan syukur kepada Allah S.W.T., karena atas rahmat-Nya, penulis dapat menyelesaikan makalah ini. Selain itu, penulis juga mengucapkan terima kasih kepada Bapak Dr. Ir. Rinaldi Munir, M.T., Ibu Dr. Nur Ulfa Maulidevi, S.T, M.Sc. dan Ibu Dr. Masayu Leylia Khodra selaku dosen pengampu kuliah IF2211 Strategi Algoritma yang telah memberikan ilmu yang bermanfaat kepada penulis.

DAFTAR PUSTAKA

- [1]. Rinaldi Munir, "Diktat Kuliah IF2211 Strategi Algoritma", Program Studi Teknik Informatika, Institut Teknologi Bandung, 2009.
- [2]. Galih Andana, "Analisis Frekuensi pada Teks Bahasa Indonesia dan Modifikasi Algoritma Kriptografi Klasik", Program Studi Teknik Informatika, Institut Teknologi Bandung, 2010.
- [3]. Alex Xandra A. S. "Algoritma Greedy", <https://bertzzie.com/knowledge/analisis-algoritma/Greedy.html>, diakses pada 17 Mei 2017 19.35 WIB.
- [4]. WikiHow, "Bermain Hangman", <http://id.wikihow.com/Bermain-Hangman>, diakses pada 17 Mei 2017 18.20 WIB.
- [5]. Cornell Math, "English Letter Frequency", <https://www.math.cornell.edu/~mec/2003-2004/cryptography/subs/frequencies.html>, diakses pada 17 Mei 2017 18.28 WIB

PERNYATAAN

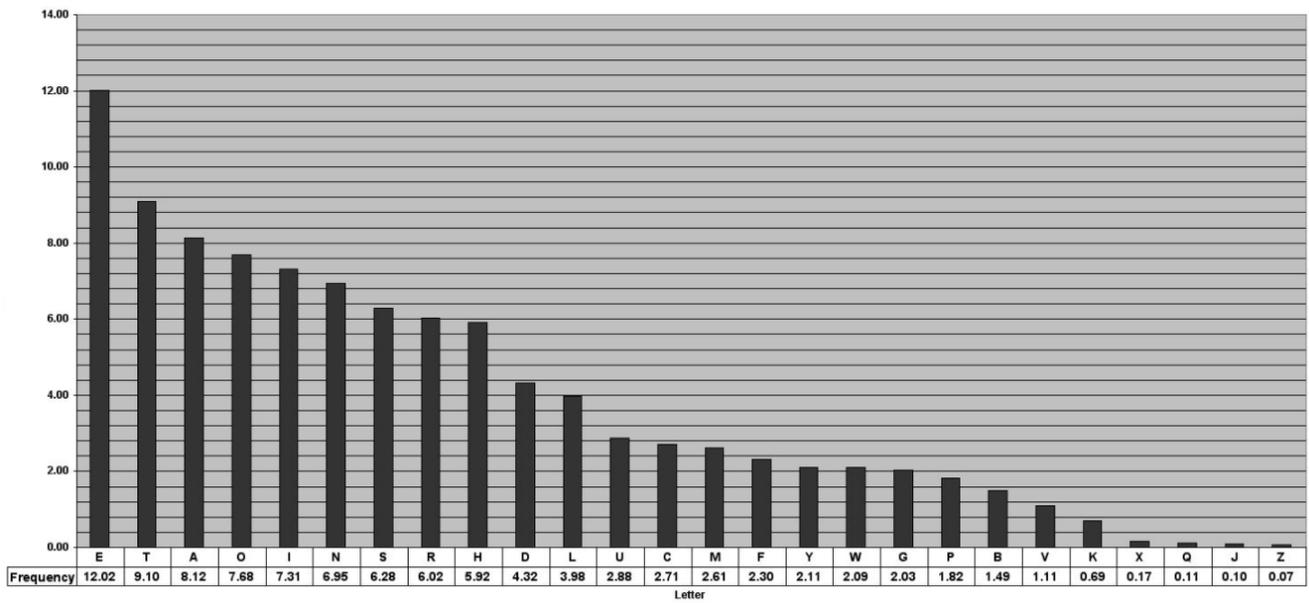
Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 Mei 2017



Diki Ardian Wirasandi
13515092

LAMPIRAN



Grafik 1.1 *English Letter Frequency*, sumber: [5]