

Algoritma Brute Force dan Pencocokan String pada Game Word Search

Cilvia Sianora Putri, 13512027
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13512027@std.stei.itb.ac.id

Abstract—Di zaman kini, dengan kerasnya perjalanan hidup, banyak orang yang bermain game untuk refreshing. Game yang dimainkan pun berbagai macam, mulai dari yang sederhana hingga yang kompleks. Namun, ada juga game sederhana yang bisa dimainkan beramai-ramai bersama keluarga atau teman seperti Word Search. Pada dasarnya untuk bermain Word Search yang digunakan adalah pencocokan string.

Index Terms—Word Search, Brute Force, String Matching

I. PENDAHULUAN

Saat ini banyak orang yang bermain game dengan berbagai alasan, seperti mengisi waktu luang, menghilangkan kebosanan, *refreshing*, dsb. Game tersebut berada di mana-mana, seperti PC, handphone, dan berbagai gadget lainnya. Game yang dimainkan pun bermacam-macam, seperti strategi, action, adventure, RPG, game shooter, racing, simulation, tycoon, arcade, dan mulai dari yang sederhana seperti tetris hingga yang kompleks seperti DotA. Dari seluruh game yang ada, salah satunya adalah Word Search, game berjenis puzzle. Word Search ini bisa dimainkan beramai-ramai bersama teman maupun keluarga.

Word Search merupakan permainan pencarian kata di dalam sebuah papan permainan berisi huruf-huruf tersusun rapi membentuk kotak. Kata-kata yang dicari bisa ditemukan dalam bentuk vertikal, horizontal, diagonal, maupun reversi-nya. Kata-kata yang perlu dicari sudah diberitahu sebelumnya. Kata-kata tersebut disembunyikan secara acak di dalam papan permainan. Kepopuleran dari permainan ini adalah dapat dilakukan bersama-sama keluarga maupun teman.

II. DASAR TEORI

II.1 Algoritma Brute Force

Algoritma brute force adalah algoritma yang *straightforward* atau menggunakan cara yang sangat sederhana dan jelas untuk memecahkan suatu permasalahan. Algoritma ini mencatat semua solusi yang ada dalam suatu masalah. Dari semua solusi tersebut,

dievaluasi satu per satu dan menyimpan solusi yang terbaik saat ini. Jika seluruh solusi sudah di-evaluasi, mengumumkan solusi terbaik yang didapat.

II.2 Algoritma Pencocokan String

Algoritma pencocokan string adalah algoritma yang digunakan untuk mencari suatu pattern di dalam teks. Algoritma pencocokan string ini memiliki tiga jenis algoritma, yaitu algoritma brute force, algoritma Knuth-Morris-Pratt (KMP), dan algoritma Boyer-Moore (BM).

Algoritma brute force adalah algoritma yang *straightforward* untuk mencari pattern yang cocok. Algoritma ini menelusuri per karakter dalam pattern dari kiri ke kanan dan membandingkannya dengan string teks.

1. Pertama-tama pattern ditaruh dengan awal karakter pattern sejajar dengan awal karakter teks.
2. Dilakukan perbandingan karakter satu per satu dari kiri ke kanan.
3. Jika terjadi ketidakcocokan karakter, pattern digeser sebanyak satu karakter.
4. Dilakukan perbandingan karakter lagi dimulai dari awal karakter pattern dengan karakter yang sejajar dalam teks.

Hal ini berlangsung terus-menerus hingga ditemukan pattern yang cocok dalam teks atau teks tidak dapat ditelusuri lagi. Contoh dari pencocokan string dengan algoritma brute force adalah sebagai berikut:

Teks : They never loses

Pattern : ever

	T	h	e	y		n	e	v	e	r		l	o	s	e	s
s=1	e	v	e	r												
s=2		e	v	e	r											
s=3			e	v	e	r										
s=4				e	v	e	r									
s=5					e	v	e	r								
s=6						e	v	e	r							
s=7							e	v	e	r						

Gambar 1. Contoh pencocokan string dengan algoritma brute force

Algoritma Knuth-Morris-Pratt (KMP) adalah algoritma yang menggunakan fungsi pinggiran dalam pencocokan string. Fungsi pinggiran adalah fungsi yang mengembalikan panjang string dari prefiks yang sama dengan sufiks-nya yang terpanjang. Fungsi pinggiran ini memberitahu berapa pergeseran terbanyak yang bisa dilakukan pada pencocokan string untuk menghindari pergeseran yang tidak berguna seperti pada algoritma brute force.

Algoritma ini, seperti algoritma brute force, menelusuri per karakter dalam pattern dari kiri ke kanan dan membandingkannya dengan string teks. Akan tetapi, jika ditemukan ketidakcocokan karakter, penggeseran pattern dilakukan berdasarkan fungsi pinggiran. Hal ini berlangsung terus-menerus hingga ditemukan pattern yang cocok dalam teks atau teks tidak dapat ditelusuri lagi.

```

procedure HitungPinggiran (input m: integer,
                        input P : array[1..m] of char,
                        output b : array[1..m] of integer)
{Menghitung fungsi pinggiran untuk pattern P}

Deklarasi
  k,q : integer

Algoritma
  b[1] <- 0
  q <- 2
  k <- 0
  for q<-2 to m do
    while ((k > 0) and (P[q] != P[k+1])) do
      k <- b[k]
    endwhile
    if P[q] = P[k+1] then
      k <- k + 1
    endif
    b[q] = k
  endfor

```

```

procedure KMP (input m,n : integer,
                input P : array[1..m] of char,
                input T : array[1..n] of char,
                output idx : integer)
{ Mencari pattern P di dalam teks T menggunakan
  algoritma KMP.

  Masukan: pattern P sepanjang m dan teks T
  sepanjang n.

  Keluaran: posisi karakter pertama pattern P
  pada teks T disimpan dalam idx. Jika tidak
  ditemukan pattern P dalam teks T, idx berisi -1
}

Deklarasi
  i,j : integer
  found : boolean
  b : array[1..m] of integer

  procedure HitungPinggiran (input m: integer,
                            input P : array[1..m] of char,
                            output b : array[1..m] of integer)
  {Menghitung fungsi pinggiran untuk pattern P}

Algoritma
  HitungPinggiran(m, P, b)
  j <- 0
  i <- 1
  found <- false

```

```

while (i <= n and not found) do

  while ((j > 0) and (P[j+1] != T[i])) do
    j <- b[j]
  endwhile

  if P[j] = T[i] then
    j <- j + 1
  endif

  if j = m then
    found <- true
  else
    i <- i + 1
  endif

endwhile

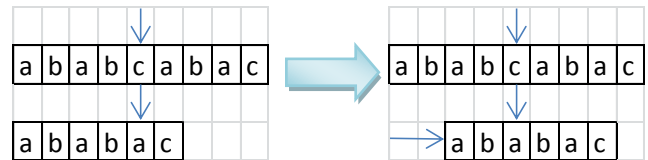
if found then
  idx <- i - m + 1 {jika indeks array mulai
  dari 1}
else
  idx <- -1
endif

```

Berikut adalah contoh pencocokan string dengan algoritma KMP.

j	1	2	3	4	5	6
P[j]	a	b	a	b	a	c
b(j)	0	0	1	2	3	0

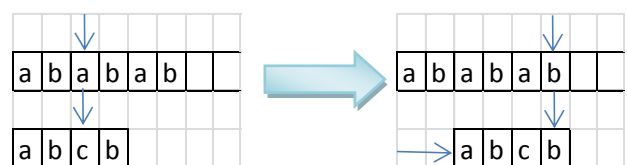
Tabel 1. Contoh hasil fungsi pinggiran



Gambar 2. Contoh pencocokan string dengan algoritma KMP

Algoritma ketiga dalam pencocokan string adalah algoritma Boyer-Moore (BM) ini menelusuri per karakter dalam pattern dari kanan ke kiri dan membandingkannya dengan string teks. Jika terjadi ketidakcocokan karakter, dilakukan penggeseran pattern ke kanan. Terdapat tiga kasus penggeseran pattern dalam algoritma ini.

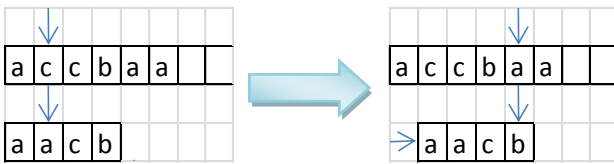
Kasus pertama adalah jika terjadi ketidakcocokan karakter dan huruf pada teks yang terkanan dalam pattern berada di sebelah kiri karakter terkini, dilakukan penggeseran pattern ke kanan hingga huruf pada teks yang terkanan dalam pattern sejajar dengan huruf pada teks yang tadi.



Gambar 3. Kasus 1 dalam algoritma Boyer-Moore

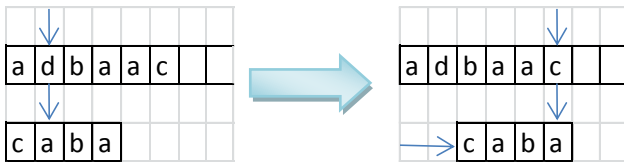
Kasus kedua adalah jika terjadi ketidakcocokan karakter dan huruf pada teks yang terkanan dalam pattern

berada di sebelah kanan karakter terkini, dilakukan penggeseran pattern ke kanan sebanyak satu karakter.



Gambar 4. Kasus 2 dalam algoritma Boyer-Moore

Kasus ketiga adalah jika terjadi ketidakcocokan karakter dan huruf pada teks tersebut tidak ada dalam pattern, dilakukan penggeseran pattern ke kanan hingga karakter pertama pattern sejajar dengan karakter setelah huruf pada teks yang tadi.



Gambar 4. Kasus 3 dalam algoritma Boyer-Moore

III. PENERAPAN DALAM WORD SEARCH

Pertama-tama, dari list kata yang ada dibuat list huruf pertama yang berisi huruf-huruf pertama dari list kata yang akan dicari.

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

ROOIBOS	YERBA	BREAKFAST
VALERIAN	ELEUTHERO	TULSI
CHAMOMILE	GINSENG	

Gambar 5. Pencarian kata pada Word Search

List huruf awal:

b	c	e	g	r	t	v	y
---	---	---	---	---	---	---	---

Pencarian kata dilakukan perbaris dari atas ke bawah dan per karakter dari kiri ke kanan, serta dimulai dari

huruf sebelah kiri atas. Jika menemukan huruf yang termasuk dalam list huruf awal, dilakukan pencocokan string.

1	2	3								
N	Y	C	H	A	M	O	M	I	L	E
I	S	L	U	T	B	V	D	J	L	N
T	S	A	F	K	A	E	R	B	E	A
Y	B	X	G	R	A	Z	A	F	L	I
A	S	F	W	B	B	G	M	V	E	R
K	A	O	Z	R	R	W	Q	Q	U	E
Q	N	N	B	B	E	H	J	M	T	L
E	O	Y	B	I	Y	I	Y	S	H	A
Z	C	E	L	B	O	X	Y	T	E	V
X	N	B	T	O	S	O	P	W	R	B
G	N	E	S	N	I	G	R	P	O	N

ROOIBOS	YERBA	BREAKFAST
VALERIAN	ELEUTHERO	TULSI
CHAMOMILE	GINSENG	

Gambar 6. Pencarian kata pada Word Search

Pencocokan string dilakukan dengan menggabungkan huruf awal yang ditemukan sebelumnya dengan huruf-huruf di sekitarnya (atas, kanan-atas, kanan, kanan-bawah, bawah, kiri-bawah, kiri, kiri-atas), lalu mencocokkannya dengan kata yang berada dalam list kata yang diawali dengan huruf awal tersebut. Dalam kasus ini berarti mencocokkan "CH", "CU", "CL", "CS", "CY" dengan kata "CHAMOMILE".

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

ROOIBOS	YERBA	BREAKFAST
VALERIAN	ELEUTHERO	TULSI
CHAMOMILE	GINSENG	

Gambar 7. Pencarian kata pada Word Search

Dari kelima awalan tersebut yang cocok adalah “CH”. Setelah itu dilakukan pencocokan dengan seluruh huruf setelah ‘H’ dalam “CH” berada hingga menemui batas papan permainan.



ROOIBOS VALERIAN CHAMOMILE
 YERBA ELEUTHERO GINSENG
 BREAKFAST TULSI

Gambar 8. Pencarian kata pada Word Search

Dan ternyata pencocokan string tersebut menemukan kata “CHAMOMILE” dengan bentuk horizontal.

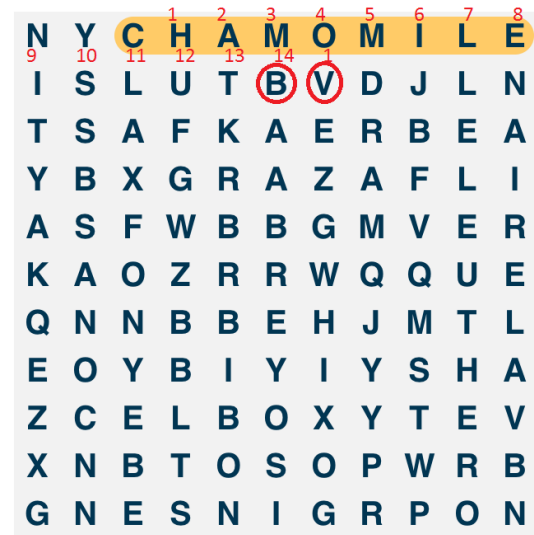


ROOIBOS VALERIAN CHAMOMILE
 YERBA ELEUTHERO GINSENG
 BREAKFAST TULSI

Gambar 9. Pencarian kata pada Word Search

Jika ternyata tidak ditemukan kata yang diperlukan atau terjadi ketidakcocokan karakter, tetap berlanjut ke tahap selanjutnya, yaitu dilakukan pencarian huruf awal lagi

seperti sebelumnya. Pencarian ini dimulai dari setelah posisi huruf awal sebelumnya ditemukan. Dalam hal ini pencarian dimulai dari ‘H’ pada baris pertama kolom keempat.



ROOIBOS VALERIAN CHAMOMILE
 YERBA ELEUTHERO GINSENG
 BREAKFAST TULSI

Gambar 10. Pencarian kata pada Word Search

Setelah dilakukan penelusuran dari ‘H’, ditemukan ‘B’ pada baris kedua. Dilakukan pencocokan string dari sekitar huruf ‘B’, yaitu “BM”, “BO”, “BV”, “BA”, “BK”, “BT”, dengan “BREAKFAST”. Akan tetapi tidak ada yang cocok sehingga dilanjutkan pencarian huruf awal lagi. Ditemukan ‘V’ pada baris kedua juga. Setelah dilakukan pencocokan string sekitar dengan “VALERIAN”, ditemukan kecocokan dengan kiri-bawah yaitu “VA”. Dilakukan pencocokan string lebih lanjut ke arah kiri-bawah dari ‘V’, yaitu “VARWONE”, dengan “VALERIAN”. Akan tetapi ditemukan ketidakcocokan pada karakter ketiga.



ROOIBOS
VALERIAN
CHAMOMILE

YERBA
ELEUTHERO
GINSENG

BREAKFAST
TULSI

Gambar 11. Pencarian kata pada Word Search

Dilakukan pencarian huruf awal lagi dimulai dari 'D' pada baris kedua. Ditemukan 'E' pada baris ketiga, tetapi dengan sekitarnya tidak memiliki kecocokan dengan "ELEUTHERO". Setelah melakukan pencarian huruf awal lagi ditemukan 'B' pada baris ketiga juga. Di sekitar 'B', sebelah kirinya, yaitu "BR", memiliki kecocokan dengan "BREAKFAST". Lalu, dilakukan pencocokan string dari "BR" dan terus ke kiri hingga batas papan permainan dengan "BREAKFAST".



ROOIBOS
VALERIAN
CHAMOMILE

YERBA
ELEUTHERO
GINSENG

BREAKFAST
TULSI

Gambar 12. Pencarian kata pada Word Search

Dan ternyata pencocokan string tersebut menemukan kata "BREAKFAST" dengan bentuk horizontal dalam

bentuk reversi atau terbalik. Semua hal ini dilakukan berulang-ulang hingga semua kata yang diperlukan ditemukan.

Setelah dilakukan berulang-ulang, seluruh kata yang diperlukan telah ditemukan. Kata-kata yang ditemukan tersebut tersembunyi dengan berbagai bentuk. Dalam contoh kasus di bawah ini "YUNNAN" ditemukan dalam keadaan diagonal reversi/terbalik dimulai di baris kesembilan, "GENMAICHA" dalam bentuk vertikal reversi/terbalik pada kolom kelima, "VALERIAN" dalam vertikal biasa pada kolom kedua, "ROOIBOS" dalam horizontal reversi/terbalik di baris terakhir, "KEEMUN" dalam horizontal biasa di baris kedua, "ECHINACEA" dalam diagonal biasa dimulai dari baris pertama, "SPEARMINT" dalam diagonal biasa dimulai dari baris kedua, dan "BREAKFAST" ditemukan di baris pertama dalam keadaan horizontal reversi/terbalik.



YUNNAN
GENMAICHA
VALERIAN

ROOIBOS
KEEMUN
ECHINACEA

SPEARMINT
BREAKFAST

Gambar 13. Seluruh kata dalam Word Search ditemukan

IV. KESIMPULAN

Game Word adalah sebuah game puzzle yang dapat diselesaikan dengan menggunakan algoritma Brute Force dan pencocokan string. Jumlah langkah yang dilakukan tergolong besar, tetapi dengan algoritma brute force dapat ditemukan seluruh kata yang diperlukan dalam bentuk vertical, horizontal, diagonal, maupun reverse-nya.

REFERENSI

- [1] Munir, Rinaldi, Diktat Kuliah IF2211 Strategi Algoritma, Program Studi Teknik Informatika Institut Teknologi Bandung, 2009.
- [2] <https://itunes.apple.com/ca/app/word-search-puzzles/id609067187?mt=8> diakses pada tanggal 16 Mei 2014
- [3] <http://imansaiki.blogspot.com/2012/03/macam-macam-genre-jenis-dalam-games.html> diakses pada tanggal 19 Mei 2014
- [4] <http://dedyjsn.wordpress.com/2008/03/24/algoritma-brute-force/> diakses pada tanggal 19 Mei 2014

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, 18 Mei 2014

A handwritten signature in black ink, appearing to read 'Cilvia Sianora Putri', written in a cursive style.

Cilvia Sianora Putri, 13512027