

# WORD SEARCH USING BOYER-MOORE ALGORITHM

Prana Pangestu<sup>1</sup>, Shinta Estri Wahyuningrum<sup>2</sup>

<sup>1,2</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Katolik  
Soegijapranata

<sup>1</sup>pranapangestu@gmail.com, <sup>2</sup>shinta@unika.ac.id

## Abstract

*Boyer-Moore is one of the algorithms used for the search words that is said to be one of the fastest to complete the search process. This project aims to implement hash table and see the effect it has in the searching process. A hash table is a table that contains key that maps to a value using a hash function. By doing so, the hash table should improve the speed of searching process. The vanilla version of Boyer-Moore will be used as the baseline to test the speed and accuracy of this method and check whether further improvement are needed.*

**Keywords:** *boyer-moore, hash table, string searching, searching, sequential search algorithm*

## Pendahuluan

Pencarian kata sering dilakukan untuk mencari kata yang dicari didalam sebuah file atau dokumen. Penelitian untuk pencarian kata perlu dilakukan agar bisa memperoleh sebuah cara yang efisien untuk mempercepat proses pencarian dan ketepatan hasil.

Ada berbagai macam algoritma untuk pencarian kata yang sering digunakan seperti brute force dan linear search. Kelebihan dari brute force adalah mudah dimengerti, bisa digunakan untuk berbagai macam masalah yang lain tetapi algoritma ini mempunyai kelemahan tidak bisa digunakan untuk permasalahan yang membutuhkan penyelesaian cepat dan cenderung lambat. Linear Search juga sering digunakan dalam proses pencarian kelebihan algoritma ini relatif cepat jika data tidak terlalu kompleks dan mudah digunakan. Kelemahannya adalah jika data terlalu banyak/kompleks maka pencarian akan berlangsung lambat.

Boyer-Moore adalah algoritma yang dianggap efisien dalam mencari kata/string karena membandingkan karakter dari sebelah kanan sehingga mempercepat proses pencarian. Jika tidak sama maka langsung bergeser ke karakter selanjutnya.

## Landasan Teori

Pada jurnal Implementasi Algoritma Boyer-Moore pada Aplikasi Kamus Kedokteran Berbasis Android. Kencana Wulan Arganingrum & Seng Hansun mengimplementasikan algoritma boyer-moore pada kamus kedokteran berbasis android [1]. Boyer-Moore digunakan untuk mencari kata yang telah disimpan sebelumnya di database. Hasil akhir dari penelitian ini adalah aplikasi android kamus kedokteran.

Jurnal kedua berjudul Implementasi Algoritma Pencarian Sequential Search pada Ensiklopedia Ikan Hias Air Tawar Berbasis Android. Haerul Umam

mengimplementasikan algoritma pencarian sequential search pada data jenis ikan hias air tawar [2]. Hasil akhir dari penelitian ini adalah aplikasi android untuk ensiklopedia.

Jurnal ketiga adalah On Improving the average case of Boyer-Moore String Matching Algorithm. Zhu Rui Feng and Tadao Takaoka menjelaskan bagaimana algoritma boyer-moore bisa digunakan untuk pencarian string [3]. Jurnal ini juga menjelaskan aturan yang berlaku di algoritma boyer-moore dan cara kerja algoritma boyer-moore.

Pada e-book berjudul Algorithm (Fourth Edition). Robert Sedgewick and Kevin Wayne menjelaskan tentang implementasi hash table. E-book ini menjelaskan struktur data Hash Table, cara kerja dari Hash Table, bentuk dari Hash Table dan implementasi dalam bahasa pemrograman [4].

Jurnal keempat berjudul Approximate Multiple String Search. Robert Muth and Udi Manber membahas tentang hash table yang bisa digunakan untuk pencarian string [5].

R. S, Boyer & J. S, Moore pada jurnal The Boyer-Moore Theorem Prover and Its Interactive Enhancement menjelaskan tentang penemuan algoritma boyer-moore, cara kerja boyer-moore dan kegunaannya [6].

## **Metodologi Penelitian**

### **1. Studi Literatur**

Mencari jurnal yang berkaitan dengan string match, boyer-moore dan hash table. Mempelajari aturan dalam algoritma boyer-moore dan cara kerja boyer-moore.

### **2. Membuat Desain Program**

Langkah selanjutnya membuat desain dan flowchart program. Merancang desain program untuk Pre-Processing dari input user, membuat BadChar Table dari input kata yang dicari, dan membuat proses pencarian menggunakan Boyer-Moore.

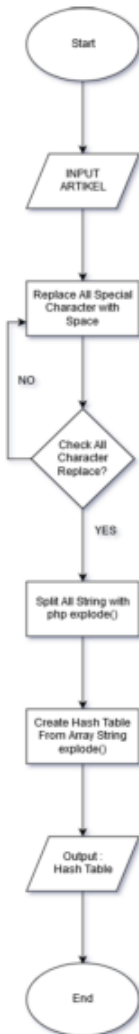
### **3. Implementasi**

Membuat program menggunakan bahasa pemrograman PHP yang digunakan untuk implementasi algoritma boyer-moore, membuat hash table, membuka dan membaca text dari input user, upload text file dan pencarian kata.

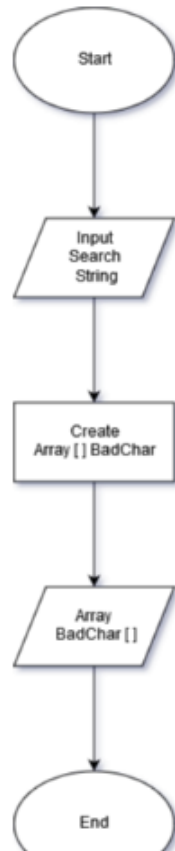
### **4. Testing:**

- a. Membandingkan boyer-moore hash table dengan boyer-moore dan membandingkan boyer-moore hash table dengan sequential search.
- b. Menganalisis hasil pencarian dan waktu eksekusi program.
- c. Mencari kata dengan algoritma Boyer-Moore Hash Table.

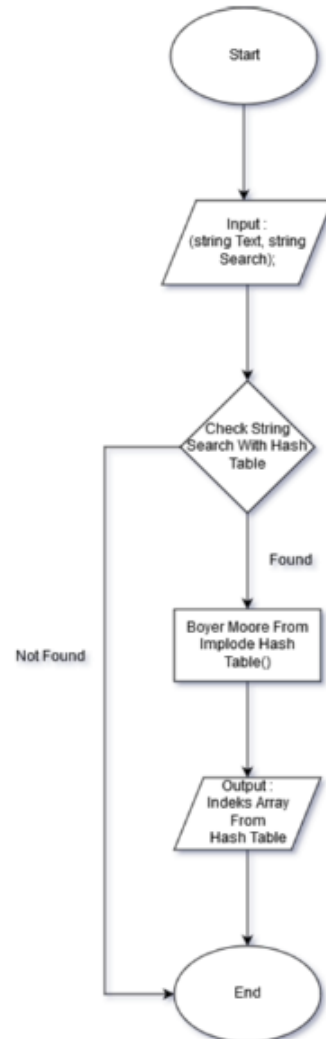
## Hasil dan Pembahasan



Gambar 1:  
Pre -  
Process



Gambar 2: Bad  
Char Table



Gambar 3: Final Process

Pada Gambar.1 menggambarkan flowchart pre-order proses sebelum menjalankan process algoritma Boyer-Moore. Pertama input file dengan format .txt kemudian semua spesial karakter di tiap string akan diganti dengan spasi. Setelah proses selesai maka dibuat hash table dari string yang telah diolah.

Setelah melakukan pre-order proses maka langkah selanjutnya membuat bad Char tabel yang digunakan untuk menghitung pergeseran pada Boyer-Moore seperti pada Gambar 2.

BadChar value = panjang dari string yang dicari – index -1

Contoh: TOOTH

T O O T H

0 1 2 3 4

Pilih semua huruf yang berbeda = TOH

Panjang string = 5

Tabel 1: Bad Char Table

Letter	T	O	H	*
Value	1	2	5	5

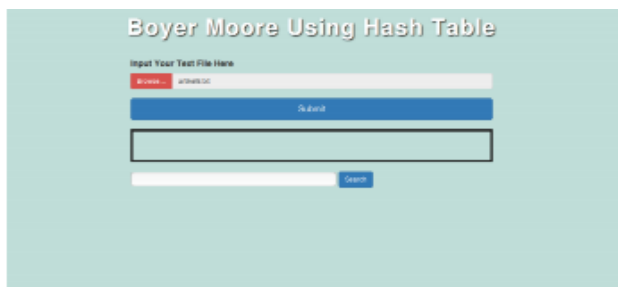
$T = 5 - 3 - 1$ ,  $O = 5 - 2 - 1$ ,  $H = 5$  (Huruf terakhir = panjang string, jika belum didefinisikan)

\* = panjang string.

Setelah Pre-Process dan BadChar sudah dilakukan maka dilanjutkan dengan proses Boyer-Moore Gambar.3. Boyer-moore dilakukan dengan memanggil function boyer-moore yang memiliki dua parameter yang berisi teks dan kata yang akan dicari. Kemudian dari kata yang dicari akan dihitung jumlah ascii dari tiap karakter nya kemudian di modulo dengan 26 untuk mendapatkan nomor indeks yang baru. Jika nomor indeks nya tidak terdapat dalam hash table yang telah dibuat maka proses akan dihentikan. Apabila nomor indeks terdaftar maka baru dilakukan proses Boyer-Moore.

Pada program tersebut terdapat form untuk upload file, tombol submit untuk menampilkan file yang telah diupload, display teks dan form pencarian kata.

1. Menampilkan nama file yang akan diupload.



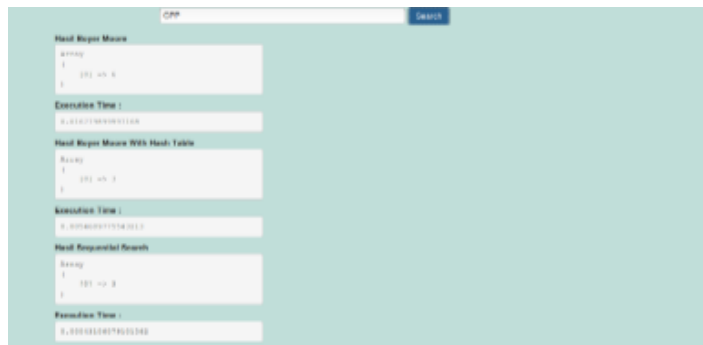
Gambar 4: upload file

2. Menampilkan hasil dari file yang telah diupload.



Gambar 5: Hasil dari teks yang telah diupload

- Mencari kata dan menampilkan hasil perbandingan serta waktu eksekusi program dari Boyer-Moore, Boyer-Moore dengan Hash Table dan Sequential Search.



Gambar 6: Hash Table dan Sequential Search

Tabel 2: Boyer-Moore

Article	String Length	String Search	Result	Execution Time
artikel8.txt	2516	CPP	Find	0
artikel9.txt	4001	Sublime	Find	0
renungan.txt	2112	Kasih	Not find	0
renungan2.txt	1976	Paulus	find	0
artikel15.txt	5764	PHP	Not Find	0

Tabel diatas menunjukkan hasil dari percobaan menggunakan boyer-moore.

Tabel 3: Result of Boyer-Moore Hash Table

Article	String Length	String Search	Result	Execution Time
artikel8.txt	2516	CPP	Find	0.01
artikel9.txt	4001	Sublime	Find	0.01
renungan.txt	2112	Kasih	Not find	0
renungan2.txt	1976	Paulus	find	0
artikel15.txt	5764	PHP	Not Find	0.01

Tabel diatas menunjukkan hasil dari percobaan menggunakan boyer-moore dengan hash table.

Tabel 4: Result of Sequential Search

Article	String Length	String Search	Result	Execution Time
artikel8.txt	2516	CPP	Find	0
artikel9.txt	4001	Sublime	Find	0
renungan.txt	2112	Kasih	Not find	0
renungan2.txt	1976	Paulus	find	0
artikel15.txt	5764	PHP	Not Find	0

Tabel diatas menunjukkan hasil dari percobaan menggunakan sequential search.

Dari ketiga tabel tersebut menunjukkan bahwa pencarian dengan menggunakan boyer-moore hash table tidak lebih cepat dari boyer-moore dan sequential search. Panjang teks dan kata yang akan dicari juga berpengaruh terhadap waktu eksekusi program.

## Kesimpulan

Kesimpulan dari project ini Boyer-Moore Hash Table tidak lebih cepat dari Boyer-Moore karena pada Boyer-Moore Hash Table text file harus dibuat menjadi hash table terlebih dahulu. Sehingga membutuhkan proses yang lebih lama. Tapi program ini lebih mudah digunakan karena menghasilkan no indeks dari kata yang dicari.

## Daftar Pustaka

- [1] K. Argakusumah and S. Hansun, "Implementasi Algoritma Boyer-Moore pada Aplikasi Kamus Kedokteran Berbasis Android," *ULTIMATICS*, vol. 6, p. 70, Dec. 2014, doi: 10.31937/ti.v6i2.340.
- [2] H. Umam, S. Hardienata, and A. Chairunnas, "Implementasi Algoritma Pencarian Sequential Search pada Ensiklopedia Ikan Hias Air Tawar berbasis Android."
- [3] F. Rui and TakaokaT, "On improving the average case of the Boyer-Moore string matching algorithm," *Journal of Information Processing*, Jul. 1988, Accessed: Feb. 04, 2021. [Online]. Available: <https://dl.acm.org/doi/abs/10.5555/50803.50808>.
- [4] R. Sedgewick and K. Wayne, *Algorithms, 4th Edition*. Addison-Wesley, 2011.
- [5] R. Muth and U. Manber, "Approximate multiple string search," in *Combinatorial Pattern Matching*, Jun. 1996, pp. 75–86, doi: 10.1007/3-540-61258-0\_7.
- [6] R. S. Boyer, M. Kaufmann, and J. S. Moore, "The Boyer-Moore theorem prover and its interactive enhancement," *Computers & Mathematics with Applications*, vol. 29, no. 2, pp. 27–62, Jan. 1995, doi: 10.1016/0898-1221(94)00215-7.