

ANALISIS PERBANDINGAN ALGORITMA BOYER MOORE,
TURBO BOYER MOORE, DAN TUNED BOYER MOORE
TERHADAP PENCARIAN STRING PADA DOKUMEN BAHASA
INDONESIA

*Diajukan Sebagai Syarat Untuk Menyelesaikan
Pendidikan Program Strata-1 Pada
Jurusan Teknik Informatika Fakultas Ilmu Komputer
Universitas Sriwijaya*



Oleh :

WIDYA DAMAYANTI

NIM : 09021181320014

Jurusan Teknik Informatika

FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA

2020

LEMBAR PENGESAHAN TUGAS AKHIR

**ANALISIS PERBANDINGAN ALGORITMA BOYER MOORE,
TURBO BOYER MOORE, DAN TUNED BOYER MOORE
TERHADAP PENCARIAN STRING PADA DOKUMEN BAHASA
INDONESIA**


Oleh :

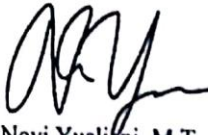
WIDYA DAMAYANTI
NIM : 09021181320014

Palembang, 1 September 2020

Pembimbing I

Pembimbing II,


Rifikie Primartha, M.T.
NIP. 197706012009121004


Novi Yusliani, M.T.
NIP. 198211082012122001

Mengetahui,

Ketua Jurusan Teknik Informatika,



Rifikie Primartha, M.T.
NIP. 197706012009121004

ii

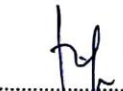
TANDA LULUS UJIAN SIDANG TUGAS AKHIR

Pada hari Selasa tanggal 20 Juli 2020 telah dilaksanakan Ujian Sidang Tugas Akhir oleh Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.

Nama : Widya Damayanti
NIM : 09021181320014
Judul : Analisis Perbandingan Algoritma *Boyer Moore*, *Turbo Boyer Moore*,
Dan *Tuned Boyer Moore* Terhadap Pencarian String Pada Dokumen
Bahasa Indonesia

1. Pembimbing I

Rifkie Primartha, M.T.
NIP. 197706012009121004



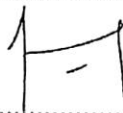
2. Pembimbing II

Novi Yusliani, S.Kom., M.T.
NIP. 198806282018031001



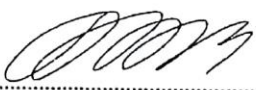
3. Penguji I

M. Fachrurrozi, S.Si., M.T.
NIP. 197812222006042003



4. Penguji II

Danny Matthew Saputra, M.Sc
NIP. 198505102015041002



Mengetahui,
Ketua Jurusan Teknik Informatika



Rifkie Primartha, M.T.
NIP. 197706012009121004

HALAMAN PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : Widya Damayanti

NIM : 09021181320014

Program Studi : Teknik Informatika

Judul Skripsi : Analisis Perbandingan Algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* Terhadap Pencarian String Pada Dokumen Bahasa Indonesia

Hasil Pengecekan Software *iThenticate/Turnitine* : 6%


Menyatakan bahwa Laporan Projek saya merupakan hasil karya sendiri dan bukan hasil penjiplakan/plagiat. Apabila ditemukan unsur penjiplakan/plagiat dalam laporan projek ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian, pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.



Indralaya, September 2020




Widya Damayanti
NIM. 09021181320014

MOTTO

"Hanya ada dua pilihan untuk memenangkan kehidupan: keberanian, atau keikhlasan. Jika tidak berani, ikhlaslah menerimannya. jika tidak ikhlas, beranilah mengubahnya".
(*Lenang Manggala*)

Kupersembahkan karya tulis ini kepada:

- ❖ Allah SWT
- ❖ Orang Tuaku
- ❖ Saudara perempuanku
- ❖ Keluarga Besarku
- ❖ Dosen Pembimbingku

Comparative Analysis of the Boyer Moore Algorithm, Turbo Boyer Moore, and Tuned Boyer Moore Against String Searching in Indonesian Documents

Widya Damayanti (09021181320014)

**Informatic Engineering, Department of Computer Science
Sriwijaya University**

ABSTRACT

Booyer Moore, Turbo Boyer Moore, and Tuned Boyer Moore algorithms are algorithms used for string searching. The Turbo Boyer Moore algorithm, and Tuned Boyer Moore are the result of the development of the Boyer Moore algorithm. These three algorithms have differences in which Boyer Moore applies the principle of the right scan rule and the shift value used is taken from the good suffix shift and bad character shift tables. Turbo Boyer Moore applies the turbo shift step, while the Tuned Boyer Moore principle works almost the same as Boyer Moore, but the shift value used is only taken from the bad character shift table. To see the performance of the three algorithms, a study of the comparison of Booyer Moore, Turbo Boyer Moore, and Tuned Boyer Moore Algorithms was done by considering accuracy, computational time, and algorithm complexity. Based on the test results it was known that Turbo Boyer Moore algorithm is better than Booyer Moore and Tuned Boyer Moore.

Keyword : Booyer Moore, Turbo Boyer Moore, Tuned Boyer Moore

Analisis Perbandingan Algoritma Boyer Moore, Turbo Boyer Moore, Dan Tuned Boyer Moore Terhadap Pencarian String Pada Dokumen Bahasa Indonesia

**Widya Damayanti (09021181320014)
Jurusan Teknik Informatika, Fakultas Ilmu Komputer
Universitas Sriwijaya**

ABSTRAK

Proses pencarian kata dalam dokumen yang cukup besar dapat menghabiskan waktu sehingga diperlukan suatu teknik pencarian yang efisien. Salah satu algoritma pencarian string yang paling baik adalah algoritma Boyer Moore. *Turbo Boyer Moore*, dan *Tuned Boyer Moore* merupakan hasil pengembangan dari algoritma *Boyer Moore*. Untuk melihat kinerja dari ketiga algoritma tersebut maka dilakukan penelitian tentang perbandingan Algoritma *Booyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* dengan mempertimbangkan akurasi, waktu komputasi, dan kompleksitas algoritma. Berdasarkan hasil pengujian tersebut diketahui bahwa algoritma *Turbo Boyer Moore* lebih baik dari *Booyer Moore* dan *Tuned Boyer Moore*.

Kata Kunci : *Booyer Moore, Turbo Boyer Moore, Tuned Boyer Moore*

KATA PENGANTAR

Puji syukur kehadirat Allah SWT, karena atas berkat, rahmat, dan karunia-Nya penulis dapat menyelesaikan Laporan Tugas Akhir yang berjudul “**Analisis Perbandingan Algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* Terhadap Pencarian String Pada Dokumen Bahasa Indonesia**”. Tugas Akhir ini disusun untuk memenuhi salah satu persyaratan kelulusan tingkat sarjana pada jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.

Pada kesempatan ini penulis mengucapkan terima kasih kepada semua pihak yang telah membantu, membimbing, dan terus mendukung penulis dalam rangka menyelesaikan laporan Tugas Akhir, di antaranya:

1. Orang Tuaku yang selalu mendoakan serta memberikan dukungan baik moril maupun materil Bapak Ismet Sastrawirya dan Ibu Masdawati, saudara perempuanku Metty Oktaviani Amd.AK dan Gladys Annisa Sastrawati;
2. Bapak Jaidan Jauhari, M.T selaku Dekan Fakultas Ilmu Komputer Universitas Sriwijaya;
3. Bapak Rifkie Primartha, M.T selaku dosen pembimbing I dan Ketua Jurusan Teknik Informatika dan Ibu Novi Yusliani, M.T. selaku pembimbing II Tugas Akhir yang telah memberikan pengarahan, bimbingan, bantuan serta masukan kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik;
4. Bapak Ir. M. Ihsan Jambak, M.Sc, Bapak M. Fachrullozzi, M.T., Bapak Danny Matthew Saputra, M.Sc selaku penguji yang telah memberikan kritik dan saran yang membangun.

5. Seluruh Dosen Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.
6. Seluruh Karyawan dan Karyawati Fasilkom UNSRI yang telah membantu dalam urusan administrasi selama kegiatan akademik.
7. Sahabat Penulis Ezil Desfa, Meita Jayani, M. Agung Multazam, Robin, Suwar, Nandes, Arif, Rudi, Yuda, dan Sandi, yang selama ini telah membantu dalam masa perkuliahan.
8. Adik-Adikku Tercinta, Meriana Ardila Putri, Varas Shafa, Rebecca Pearly Tannia yang selalu ada disetiap keadaan senang dan memalukan dan selalu menghibur dalam kondisi apapun.
9. Untuk semua pihak yang telah membantu menyelesaikan tugas akhir ini yang tidak dapat disebutkan satu persatu.

Penulis menyadari dalam penyusunan Tugas Akhir ini masih terdapat banyak kekurangan dikarenakan keterbatasan pengetahuan dan pengalaman, oleh karena itu kritik dan saran yang membangun sangat diharapkan untuk kemajuan penelitian selanjutnya. Semoga Tugas Akhir ini dapat berguna dan bermanfaat bagi kita semua.

Indralaya, September 2020

Penulis

DAFTAR ISI

| | Halaman |
|--------------------------------|------------|
| HALAMAN JUDUL | i |
| LEMBAR PENGESAHAN | ii |
| DAFTAR ISI..... | iii |
| DAFTAR TABEL | v |
| DAFTAR GAMBAR..... | vi |

BAB I PENDAHULUAN

| | |
|----------------------------------|-----|
| 1.1 Latar Belakang Masalah | I-1 |
| 1.2 Rumusan Masalah | I-3 |
| 1.3 Tujuan Penelitian..... | I-3 |
| 1.4 Manfaat Penelitian..... | I-4 |
| 1.5 Batasan Masalah..... | I-4 |

BAB II TINJAUAN PUSTAKA

| | |
|---|------|
| 2.1 Penelitian Terdahulu | II-1 |
| 2.2 Pencocokan String (<i>String Matching</i>)..... | II-1 |
| 2.3 Tahap Praproses | II-2 |
| 2.4 <i>Good-suffix-rule</i> | II-3 |
| 2.5 <i>Bad-character-rule</i> | II-3 |
| 2.6 <i>Boyer-Moore</i> | II-8 |
| 2.7 <i>Turbo Boyer-Moore</i> | II-9 |

| | |
|---|-------|
| 2.8 <i>Tuned Booyer-Moore</i> | II-11 |
| 2.9 Kompleksitas Algoritma | II-12 |
| 2.9.1 Model Perhitungan Waktu dan Ruang | II-13 |

BAB III METODOLOGI PENELITIAN

| | |
|--|--------|
| 3.1 Unit Penelitian | III-1 |
| 3.2 Metode Pengumpulan Data | III-1 |
| 3.2.1 Jenis Data | III-1 |
| 3.2.2 Sumber Data | III-1 |
| 3.2.3 Teknik Pengumpulan Data | III-1 |
| 3.3 Tahapan Penelitian | III-2 |
| 3.3.1 Melakukan Kajian Pustaka | III-2 |
| 3.3.2 Memilih Literatur yang Relevan dan Merumuskan Masalah | III-2 |
| 3.3.3 Menetapkan Kriteria Pengujian | III-2 |
| 3.3.4 Menentukan Alat yang Digunakan dalam Pelaksanaan Penelitian | III-5 |
| 3.3.5 Melakukan Pengujian Penelitian | III-5 |
| 3.3.6 Melakukan Analisa Hasil Pengujian dan Membuat Kesimpulan | III-7 |
| 3.4 Metode Pengembangan Perangkat Lunak | III-8 |
| 3.5 Penjadwalan Tugas Akhir | III-11 |

BAB IV METODOLOGI PENELITIAN

| | | |
|---------|---|-------|
| 4.1 | Pendahuluan | IV-1 |
| 4.2 | Fase Insepsi | IV -1 |
| 4.2.1 | Pemodelan Bisnis..... | IV-2 |
| 4.2.2 | Kebutuhan..... | IV-2 |
| 4.2.2.1 | Fitur Masukkan Data | IV-2 |
| 4.2.2.2 | Fitur Pencarian String | IV-2 |
| 4.2.3 | Analisis dan Desain | IV-4 |
| 4.2.3.1 | Analisis Kebutuhan Perangkat Lunak | IV-4 |
| 4.2.3.2 | Analisis Data..... | IV-4 |
| 4.2.3.3 | Analisis Fitur Pencarian..... | IV-5 |
| 4.2.3.4 | Analisis Prapengolahan | IV-7 |
| 4.2.3.5 | Analisis Algoritma <i>Boyer Moore</i> | IV-9 |
| 4.2.3.5 | Analisis Algoritma <i>Turbo Boyer Moore</i> | IV-12 |
| 4.2.3.7 | Analisis Algoritma <i>Tuned Boyer Moore</i> | IV-13 |
| 4.2.3.8 | Desain Perangkat Lunak | IV-16 |
| 1. | Model <i>Use Case</i> | IV-16 |
| a. | Diagram <i>Use Case</i> | IV-16 |
| b. | Tabel Definisi Aktor..... | IV-17 |
| c. | Tabel Definisi <i>Use Case</i> | IV-18 |
| d. | Skenario <i>Use Case</i> | IV-18 |
| 2. | Diagram Aktivitas | IV-20 |
| 4.3 | Fase Elaborasi | IV-22 |

| | | |
|---------|---|-------|
| 4.3.1 | Pemodelan Bisnis | IV-23 |
| 4.3.1.1 | Perancangan Data | IV-23 |
| 4.3.1.2 | Perancangan Antar Muka | IV-23 |
| 4.3.2 | Kebutuhan Sistem | IV-25 |
| 4.3.3 | Diagram <i>Sequence</i> | IV-25 |
| 4.4 | Fase Konstruksi | IV-28 |
| 4.4.1 | Pemodelan Bisnis | IV-28 |
| 4.4.2 | Kebutuhan | IV-28 |
| 4.4.3 | Implementasi | IV-29 |
| 4.4.3.1 | Implementasi Kelas | IV-29 |
| 4.4.3.2 | Implementasi Antarmuka | IV-32 |
| 4.5 | Fase Transisi | IV-32 |
| 4.5.1 | Pemodelan Bisnis | IV-33 |
| 4.5.2 | Kebutuhan | IV-33 |
| 4.5.3 | Analisis dan Desain | IV-33 |
| 4.5.3.1 | Rencana pengujian <i>Use Case</i> Memasukkan Data | IV-34 |
| 4.5.3.2 | Rencana Pengujian <i>Use Case</i> Melakukan Pencarian String dengan <i>Boyer Moore</i> | IV-34 |
| 4.5.4 | Implementasi | IV-35 |
| 4.5.4.1 | Pengujian <i>Use Case</i> Memasukkan Data | IV-35 |
| 4.5.4.2 | Pengujian <i>Use Case</i> Melakukan Pencarian String | IV-35 |
| 4.6 | Kesimpulan | IV-37 |

BAB V IMPLEMENTASI DAN PENGUJIAN

| | | |
|-------|---|------|
| 5.1 | Pendahuluan | V-1 |
| 5.2 | Data Hasil Pengujian Algoritma <i>Boyer Moore</i> | V-2 |
| 5.2.1 | Data Hasil Uji Pencarian String <i>Boyer Moore</i> | V-2 |
| 5.3 | Data Hasil Pengujian <i>Turbo Boyer Moore</i> | V-15 |
| 5.3.1 | Data Hasil Uji Pencarian String <i>Turbo Boyer Moore</i> | V-15 |
| 5.4 | Data Hasil Pengujian <i>Tuned Boyer Moore</i> | V-24 |
| 5.4.1 | Data Hasil Uji Pencarian String <i>Tuned Boyer Moore</i> | V-24 |
| 5.5 | Tingkat Akurasi | V-33 |
| 5.6 | Waktu Komputasi | V-33 |
| 5.7 | Kompleksitas Waktu Asimptotik Algoritma | V-34 |
| 5.7.1 | Kompleksitas Waktu Asimptotik Algoritma <i>Boyer Moore</i> , <i>Turbo Boyer Moore</i> , dan <i>Tuned Boyer Moore</i> | V-34 |
| 5.8 | Analisis Penelitian | V-37 |

BAB VI KESIMPULAN DAN SARAN

| | | |
|-----|-------------------|------|
| 6.1 | Pendahuluan | VI-1 |
| 6.2 | Kesimpulan..... | VI-2 |
| 6.3 | Saran..... | VI-3 |

| | |
|-----------------------------|-----|
| DAFTAR PUSTAKA | xii |
|-----------------------------|-----|

DAFTAR TABEL

| | Halaman |
|--|---------|
| Tabel III-1 Rancangan Tabel Hasil Pengujian | III-4 |
| Tabel III-2 Tabel Pengembangan Perangkat Lunak..... | III-10 |
| Tabel IV-1. Kebutuhan Fungsional | IV-3 |
| Tabel IV-2. Kebutuhan Non Fungsional | IV-3 |
| Tabel IV-3 Nilai <i>Ocurrence Heuristic</i> | IV-8 |
| Tabel IV-4 Nilai <i>Math Heuristic</i> | IV-9 |
| Tabel IV-5 Pendekatan <i>Occurance Heuristic</i> | IV-10 |
| Tabel IV-6 Pendekatan <i>Match Heuristic</i> | IV-10 |
| Tabel IV-7 Pencocokan Karakter ke-1 | IV-11 |
| Tabel IV-8 Pencocokan Karakter ke-1 | IV-11 |
| Tabel IV-9 Pencocokan Karakter ke-2 | IV-12 |
| Tabel IV-10 Hasil Pencocokan karakter | IV-12 |
| Tabel IV-11 Pencocokan Karakter 1 | IV-13 |
| Tabel IV-12 Pencocokan Karakter 1 | IV-14 |
| Tabel IV-13. Pergeseran <i>Occurance Heuristic</i> | IV-15 |
| Tabel IV-14. Pencocokan Karakter ke-1 | IV-15 |
| Tabel IV-15. Pencocokan Karakter ke-1 | IV-15 |
| Tabel IV-16. Pencocokan Karakter ke-1 | IV-16 |
| Tabel IV-17. Definisi Pengguna <i>Use Case</i> | IV-17 |
| Tabel IV-18. Definisi <i>Use Case</i> | IV-18 |
| Tabel IV-19 Skenario Memasukkan Data | IV-19 |

| | |
|---|-------|
| Tabel IV-20. Tabel Implementasi Kelas | IV-30 |
| Tabel IV-21. Rencana Pengujian <i>Use Case</i> Memasukkan Data | IV-34 |
| Tabel IV-23. Rencana Pengujian <i>Use Case</i> Melakukan Pencarian String dengan <i>Boyer Moore, Turbo Boyer Moore, dan Tuned Boyer Moore</i> 33 | IV-34 |
| Tabel IV-24. Pengujian <i>Use Case</i> Memasukkan Data | IV-35 |
| Tabel IV-25. Pengujian <i>Use Case</i> Melakukan Pencarian String | IV-36 |
| Tabel V-1 Hasil Pencarian String <i>Boyer Moore</i> dengan Jumlah Huruf Berbeda..... | V-3 |
| Tabel V-2. Hasil Pencarian String <i>Turbo Boyer Moore</i> dengan Jumlah Huruf Berbeda 16 | V-16 |
| Tabel V-3. Hasil Pencarian String <i>Tuned Boyer Moore</i> dengan Jumlah Huruf Berbeda | V-25 |
| Tabel V-4. Akurasi Pencarian String | V-33 |
| Tabel V-5. Rata-Rata Waktu Komputasi | V-34 |
| Tabel V-6. Perbandingan Kompleksitas Waktu Pada Proses Pencarian | V-34 |

DAFTAR GAMBAR

| | Halaman |
|---|---------|
| Gambar II-1 <i>Good-suffix-rule</i> | II-2 |
| Gambar II-2 <i>Bad-character -rule</i> | II-3 |
| Gambar IV-1. Alur Pencarian String pada Perangkat Lunak | IV-5 |
| Gambar IV-2. Diagram <i>Use Case</i> | IV-17 |
| Gambar IV-3 Diagram Aktivitas <i>Use Case</i> Memasukkan Data | IV-22 |
| Gambar IV-4 Diagram Aktivitas <i>Use Case</i> Melakukan Pencarian String | IV-22 |
| Gambar IV-5 Rancangan Antar Muka Perangkat Lunak | IV-24 |
| Gambar IV-6 <i>Sequence</i> Diagram Masukkan Data | IV-26 |
| Gambar IV-7 <i>Sequence</i> Diagram Melakukan Pencarian String | IV-27 |
| Gambar IV-8 Diagram Kelas | IV-29 |
| Gambar IV-10 Antar muka Utama Perangkat Lunak | IV-32 |
| Gambar V-1 Perbandingan Waktu Komputasi Pencarian String Algoritma <i>Boyer Moore, Turbo Boyer Moore, dan Tuned Boyer Moore</i> | IV-38 |
| Gambar V-2 Perbandingan Kompleksitas Waktu Asimptotik Algoritma <i>Boyer Moore, Turbo Boyer Moore, dan Tuned Boyer Mooore</i> | IV-39 |

BAB I

PENDAHULUAN

1.1 Pendahuluan

Bab 1 pada penelitian ini membahas latar belakang, rumusan masalah, tujuan penelitian, dan batasan masalah. Bab ini menjelaskan mengenai latar belakang metode yang digunakan untuk pencarian string dalam dokumen.

1.2 Latar Belakang Masalah

Data berupa teks merupakan data yang paling umum digunakan untuk pertukaran informasi dan manfaat lainnya. Sekarang ini teks sudah disimpan dalam bentuk digital, hal ini semakin mempermudah pertukaran dan penggandaan teks sehingga jumlah teks semakin banyak. Banyaknya jumlah kata dalam sebuah dokumen menyebabkan adanya suatu kebutuhan untuk mencari suatu kata dalam dokumen tersebut. Pencarian kata di dalam dokumen dilakukan untuk mencari kata yang tepat sesuai kebutuhan pengguna dan diharapkan dapat menuntaskan pekerjaannya dengan cepat.

Berdasarkan arah pencocokan string, sistem pencarian string dapat dibagi menjadi tiga yaitu dari kiri ke kanan, kanan ke kiri, dan dari arah yang ditentukan secara spesifik (Kumara, 2009). Kebanyakan penelitian dilakukan dengan menggunakan algoritma pencocokan string dari arah kiri ke kanan karena sama seperti arah membaca.

Penelitian dengan pencocokan string dari kanan ke kiri telah dilakukan sebelumnya menggunakan algoritma *Boyer Moore* (Noorfaik dan Novianto, 2014) memiliki akurasi sebesar 86%. Penelitian selanjutnya dengan membandingkan dua

algoritma, yaitu penelitian dengan membandingkan algoritma *Knuth-Morris-Pratt* dan *Karp-Rabin* (Fernando, 2010), penelitian ini menyimpulkan bahwa algoritma *Knuth-Morris-Pratt* lebih cepat dalam pencocokan string ketika *pattern* yang dicari hanya satu atau tunggal, sedangkan penelitian dengan membandingkan dua algoritma yaitu algoritma *Booyer Moore* dan *Karp Rabin* (Utomo dan Handoko, 2008) menyimpulkan bahwa algoritma *Booyer Moore* lebih cepat dalam pencarian teks dibandingkan dengan algoritma *Karp Rabin*.

Boyer-Moore merupakan algoritma yang paling efektif dan efisien dalam kehidupan sehari-hari (Wibowo, 2012). Kelebihan utama algoritma *Boyer-Moore* adalah membaca karakter-karakter dari *pattern* kanan ke kiri sehingga informasi yang didapat lebih cepat (Aulia, 2008). Algoritma *Boyer-Moore* dalam pencarian kata digunakan untuk pencocokan *string* antara kata yang dicari dengan kata yang ada didalam teks. Algoritma *Boyer Moore* kemudian dikembangkan menjadi algoritma *Turbo Boyer-Moore* dan *Tuned Boyer-Moore*.

Turbo Boyer-Moore merupakan variasi dari algoritma *Boyer-Moore*. Bila dibandingkan dengan algoritma *Boyer-Moore*, algoritma ini membutuhkan ruang lebih tapi tidak memerlukan pemrosesan ekstra (Charras dan Lecroq, 2009). Ruang ekstra yang diperlukan berguna untuk mengingat faktor dari teks yang cocok dengan akhiran dari *string* yang dicari selama *attempt* terakhir dan hanya jika *good-suffix shift* dilakukan (Charras dan Lecroq, 2009).

Tuned Boyer-Moore merupakan simplifikasi dari algoritma *Boyer-Moore*, mudah untuk diimplementasikan, hanya menggunakan *bad-character shift*, dan sangat cepat dalam praktiknya (Lee & Cheng, 2007). Pada dasarnya dalam

algoritma ini yang menjadi fokus utama adalah karakter terakhir dari *pattern* dan menggeser *pattern* untuk mencocokkan bagian paling terakhir dari *pattern* tersebut (Lee & Cheng, 2007).

Berdasarkan kelebihan dan kekurangan dari algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned boyer Moore* yang telah diuraikan sebelumnya, maka pada penelitian ini akan dilakukan perbandingan algoritma *Boyer Moore*, *Turbo Boyer-Moore*, dan *Tuned Boyer-Moore* untuk kasus pencarian string dokumen bahasa Indonesia.

1.3 Rumusan Masalah

Perumusan masalah yang diangkat dari penelitian ini adalah bagaimana perbandingan kinerja metode *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* dalam pencarian string.

Untuk menyelesaikan permasalahan yang disebutkan diatas, maka dalam penelitian ini dibatasi dalam beberapa pertanyaan penelitian (*research question*) antara lain :

1. Bagaimana mekanisme metode *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* dalam pencarian string.
2. Bagaimana mengukur kinerja dari ketiga metode tersebut dalam pencarian string.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah :

1. Mengetahui kelebihan dan kekurangan algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore*.
2. Membandingkan ketiga algoritma yang memiliki akurasi paling baik dan membutuhkan waktu paling cepat dalam pencarian kata.

1.5 Manfaat Penelitian

Manfaat pada penelitian ini adalah :

1. Memahami algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* sebagai algoritma pencarian kata.
2. Mengetahui perbandingan algoritma *Boyer Moore*, *Turbo Boyer Moore*, dan *Tuned Boyer Moore* sehingga mendapatkan metode yang paling baik untuk pencarian kata.

1.6 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Masukan perangkat lunak berupa dokumen bertipe *.txt.
2. Penelitian hanya membandingkan tingkat akurasi dan kecepatan waktu dari ketiga algoritma tersebut.

Daftar Pustaka

- Aulia, R. (2008). Analisis Algoritma Knuth Morris Pratt Dan Algoritma Boyer Moore Dalam Proses Pencarian String: Bandung: Itb.
- Charras, Christian & Thierry Lecroq. 2009. "Handbook of Exact-String Matching Algorithms". URL: http://ebookey.org/Handbook-of-Exact-String-Matching-Algorithms_892540.html. Diakses tanggal 28 Februari 2012.
- Faro, S., & Lecroq, T. (2009). *An Efficient Matching Algorithm For Encoded Dna Sequences And Binary Strings*. Paper Presented At The Annual Symposium On Combinatorial Pattern Matching.
- Fernando, H. (2009). Perbandingan Dan Pengujian Beberapa Algoritma Pencocokan String. *Program Studi Teknik Informatika, Institut Teknologi Bandung (Itb). Bandung*.
- Handoko, D. U. E. W. H. (2008). Perbandingan Algoritma String Searching Brute Force, Knuth Morris Pratt, Boyer Moore, Dan Karp Rabin Pada Teks Alkitab Bahasa Indonesia.
- Hussain, I., Kausar, S., Hussain, L., & Khan, M. A. (2013). Improved Approach For Exact Pattern Matching. *Int. J. Comput. Sci. Issues*, 10, 59-65.
- Irianto, S., & Mustafidah, H. (2006). Analisis Kompleksitas Waktu Dan Ruang Terhadap Laju Pertumbuhan Algoritma Heapsort (Time And Space Complexity Analysis For Heapsort Algorithm Growth Rate). *Biomath*, 7(2).
- Jansen, Klaus, Margraf, M., Mastrolili, & Rolim. (2003). Experimental And Efficient Algorithms.
- Kristanto, S., & Santosa, R. G. (2016). Implementasi Algoritma Boyer-Moore Pada Permainan Word Search Puzzle. *Proceedings Of Knastik*.
- Kroll, P., & Kruchten, P. (2003). *The Rational Unified Process Made Easy: A Practitioner's Guide To The Rup*: Addison-Wesley Professional.
- Kumara. (2009). Visualisasi Beberapa Algoritma Pencocokan String dengan Java. *Jurnal: Jurusan Teknik Informatika, Fakultas Elektro Informatika, Institut Teknologi Bandung*.
- Lee, & Cheng. (2007). Tuned Boyer Moore Algorithm.

Noorfaik, M. & Novianto, S., (2014). Efisiensi Penggunaan Algoritma Boyer Moore Untuk Prediksi Perilaku Orang Melalui Interaksi Di Twitter. *Skripsi, Fakultas Ilmu Komputer*.

Wibowo, A. & Pratiwi, H. R., Ss, D. S., (2012). Prototype Aplikasi Sms Content Filtering Menggunakan Metode String Matching (Studi Kasus: Content