

Implementasi Algoritma *Brute Force* Dalam Pencocokan *String* Pada Aplikasi Pencarian Musik

M. Hafizh Al-Areef¹, Nazifatul Fadhilah², Muhammad Rifqi Naufal³, Muhammad Rifqi Maulana⁴, Muhammad Fajar Harahap⁵, dan Kana Saputra S⁶

^{1,2,3,4,5,6}Program Studi Ilmu Komputer, Fakultas MIPA, Universitas Negeri Medan
Jl. Willem Iskandar / Pasar V, Medan, Sumatera Utara

E-mail : hafizh2608@mhs.unimed.ac.id¹, fadhilahnazifatul@gmail.com²,
m.rifqinaufal05@gmail.com³, mhdrifqimaulana@gmail.com⁴, mhdfajar4495@gmail.com⁵,
kanasaputras@unimed.ac.id⁶

Abstract—Music is something we often encounter in everyday life. Many of our daily activities are accompanied by music. This web-based music application can assist users in searching for information about music, such as title, album, artist, and lyrics. Previously searching for song lyrics by song title, singer was a difficult thing. This study aims to create a song lyrics search application by applying the brute force algorithm as a search feature. Users just enter the song title or song lyrics or singer name to perform a search. The brute force algorithm is suitable for use in the case of string matching in searching for information about music in terms of effectiveness and time efficiency.

Abstrak—Musik adalah hal yang sering kita jumpai dalam kehidupan sehari-hari. Banyak kegiatan sehari-hari kita yang ikut ditemani dengan musik. Aplikasi musik berbasis web ini dapat membantu pengguna dalam pencarian informasi mengenai suatu musik, seperti judul, album, artis, dan liriknya. Sebelumnya pencarian lirik lagu berdasarkan judul lagu, penyanyi adalah hal yang sulit. Penelitian ini bertujuan untuk membuat aplikasi pencarian lirik lagu dengan menerapkan algoritma *brute force* sebagai fitur pencari. Pengguna hanya memasukkan judul lagu atau lirik lagu atau nama penyanyi untuk melakukan pencarian. Algoritma *brute force* cocok digunakan dalam kasus pencocokan *string* dalam pencarian informasi mengenai musik yang dilihat dari sisi efektifitas dan efisiensi waktu.

Kata Kunci—Algoritma *Brute Force*, Aplikasi, Lagu, Pencocokan *String*.

I. PENDAHULUAN

MUSIK merupakan salah satu kebutuhan pokok bagi manusia, karena musik dapat membuat manusia merasa senang, gembira, dan nyaman [1]. Lirik dari sebuah lagu merupakan ungkapan perasaan berdasarkan pengalaman, cerita, atau penglihatan seseorang. Lirik lagu merupakan media perantara untuk menyampaikan sebuah pesan, maksud dan makna dibalik lirik [2]. Secara definisi algoritma adalah alur pemikiran yang logis yang dapat dituangkan ke dalam bentuk tulisan [3]. *String* dalam bidang komputer merupakan sekuens dari karakter. Selain itu, *string* juga dianggap sebagai data abstrak yang menyimpan sekuens nilai data, atau biasanya berupa *bytes* yang mana merupakan elemen yang digunakan sebagai pembentuk karakter sesuai dengan *encoding* karakter yang disepakati seperti ASCII, ataupun EBCDIC [4]. *String matching* atau yang biasa juga disebut dengan kata pencocokan *string* adalah metode pendekatan untuk menemukan keterhubungan atau kemiripan antar dua buah yang *string*, yaitu *string pattern* dan *string text* [5]. Terdapat dua teknik dalam algoritma *string matching*, yaitu *exact matching* yang merupakan hasil pencocokannya mengandung *string* yang tidak ada bedanya dengan *string* yang di-input atau sama persis dan *approximate matching* dimana hasil pencocokannya mengandung *string* yang memiliki sedikit perbedaan atau dengan kata lain tidak harus persis dengan *string* yang diinput [6].

Saat ini *web* merupakan salah satu sumber informasi yang banyak dipakai. *Web* dibuat dengan tujuan agar

pemakai dapat berinteraksi dengan penyedia informasi secara mudah dan cepat melalui jaringan internet [7]. Informasi mengenai suatu musik yang dibutuhkan secara cepat oleh seseorang merupakan suatu permasalahan yang ingin diselesaikan. Maka dari itu adapun tujuan penelitian ini adalah untuk menerapkan algoritma *brute force* dalam perancangan aplikasi pencarian musik berbasis *website*. Algoritma *brute force* memecahkan suatu teknik yang biasa digunakan bila penyusun algoritma lebih mempertimbangkan memperoleh solusi dari *problem* secara langsung [8]–[10]. Pengguna hanya tinggal mendefinisikan karakter set yang diinginkan dan berapa ukuran dari kodenya. Tiap kemungkinan kode akan di *generate* oleh algoritma ini [11].

Algoritma *brute force* sudah pernah diimplementasikan dalam pencocokan *string* dengan hasil dapat memecahkan masalah hampir sebagian besar masalah karena alur yang sangat sederhana dan mudah dimengerti [12]. Selain itu, Algoritma *Brute force* sudah pernah diimplementasikan sebagai mesin pencari berbasis web pada data base dengan hasil memudahkan para pengguna untuk mengetahui jenis obat generic dan zat yang terkandung didalamnya tanpa terbatas ruang dan waktu [13]. Algoritma *brute force* juga pernah diimplementasikan pada perancangan aplikasi kamus dengan hasil yang cukup baik [14], [15].

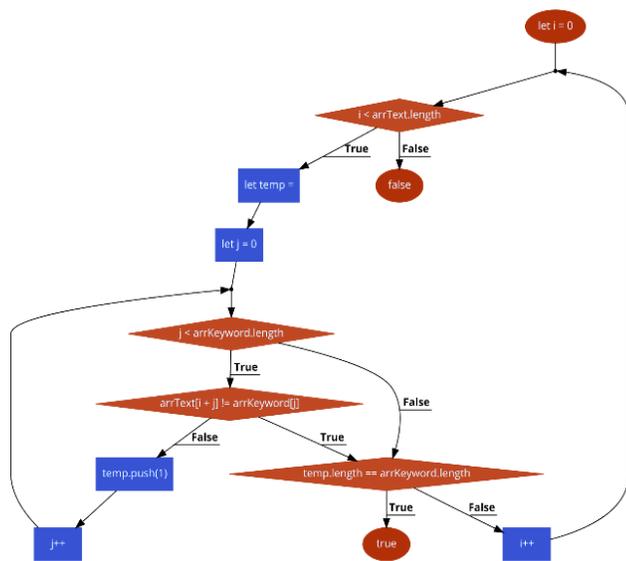
II. METODE PENELITIAN

A. Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah judul lagu, penyanyi, album dan lirik lagu. Data tersebut diperoleh dari *google* dengan pencarian secara manual dengan jumlah data sebanyak 40 lagu.

B. Implementasi Algoritma Brute Force

Algoritma *brute force* ini bekerja berdasarkan teknik *straight forward* dimana algoritma ini menggeser *pattern* di setiap karakter teks, lalu membandingkan setiap karakter pada *pattern*, sampai *pattern* itu ditemukan di teks atau terjadi ketidakcocokan. Berikut *flowchart* algoritma *brute force* yang akan diimplementasikan ke dalam fungsi pencarian musik.



Gambar. 1. Flowchart Algoritma Brute Force.

C. Desain Aplikasi Pencarian Musik

Tampilan desain aplikasi pencarian musik mempertimbangkan kemudahan dari sisi pengguna. Pada bagian fitur pencarian akan menerapkan algoritma *brute force* untuk mempermudah pengguna dalam mencari lirik lagu dengan cepat dan benar. Tampilan desain aplikasi pencarian musik dapat dilihat pada Gambar 2.



Gambar. 2. Desain Aplikasi Pencarian Musik.

III. HASIL DAN PEMBAHASAN

A. Data Musik

Data diolah didalam teks editor dan disusun dengan format judul, penyanyi, album lalu lirik. Detail sampel data musik yang diperoleh dapat dilihat pada Tabel 1.

Tabel 1.
Detail Sampel Data Musik

No	Judul	Penyanyi	Album
1	Bersamamu	Vierratale	My First Love
2	RED	Taylor Swift	Red
3	Payphone	Maroon 5	Overexposed

B. Implementasi Algoritma Brute Force

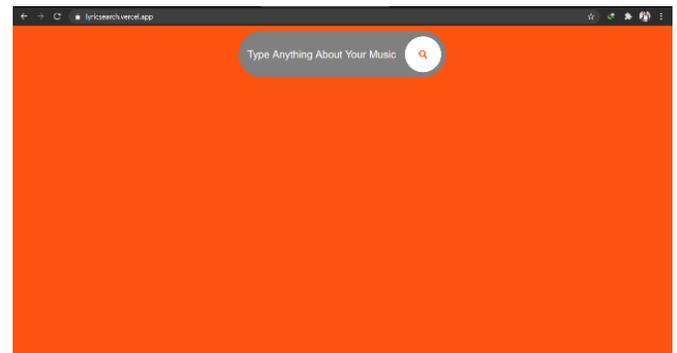
Data yang telah terkumpul akan dihubungkan kedalam script javascript yang isinya adalah *function brute force* untuk pencarian lirik, judul, album hingga penyanyi. Berikut *function brute force*:

```
function bruteForce(text, keyword) {
  const arrText = text.split("");
  const arrKeyword = keyword.split("");

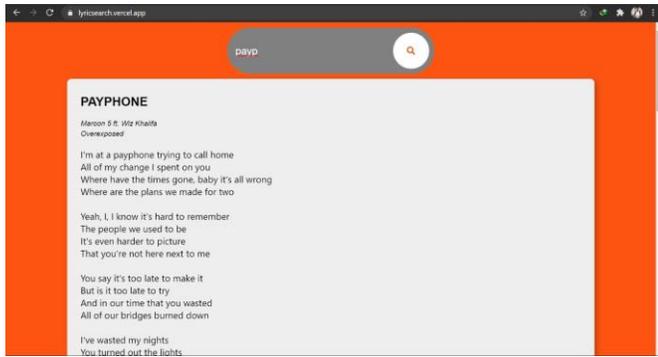
  for (let i = 0; i < arrText.length; i++) {
    let temp = [];
    for (let j = 0; j < arrKeyword.length; j++) {
      if (arrText[i + j] != arrKeyword[j]) {
        break;
      } else {
        temp.push(1);
      }
    }
    if (temp.length == arrKeyword.length) {
      return true;
    }
  }
  return false;
}
```

C. Tampilan Aplikasi Pencarian Musik

Berikut tampilan aplikasi pencarian musik. Terdapat searchbar diatasnya bagi pengguna untuk mencari informasi mengenai musik yang ingin dicari.



Gambar. 3. Tampilan Utama Aplikasi Pencarian Musik.



Gambar. 4. Tampilan Aplikasi Pencarian Musik Dengan Kata Kunci Payp.

D. Pengujian

Aplikasi yang telah dibuat kemudian dilakukan uji coba dengan beberapa scenario seperti yang terlihat pada Tabel 2.

Tabel 2. Hasil Uji coba Aplikasi Pencarian Musik

No	Kata yang dicari	Tipe	Status	Hasil Pencarian
1	Bersamamu	Judul	Ditemukan	Bersamamu
	Maroon 5	Artis	Ditemukan	Payphone, Just A Feeling
3.	The Special One	Album	Ditemukan	Menjaga Hati, Dia Milikku
4	I'm at a payphone trying to call home	Lirik	Ditemukan	I'm at a payphone trying to call home All of my change I spent on you

IV. KESIMPULAN

Dengan adanya hasil penelitian yang telah dilaksanakan serta uraian yang telah dibahas sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

- 1) Algoritma *brute force* merupakan algoritma pencarian yang dapat diterapkan dalam perancangan aplikasi pencarian musik berbasis *web* sehingga dapat mempermudah pengguna dalam mencari informasi terkait musik dalam waktu singkat.
- 2) Algoritma *brute force* cocok digunakan dalam kasus pencocokan *string* dalam pencarian informasi mengenai musik. Baik dalam efektifitas dan efisiensi waktu pencarian yang hanya membutuhkan waktu yang sangat singkat.

DAFTAR PUSTAKA

- [1] A. Roffiq, I. Qiram, and G. Rubiono, "Media Musik Dan Lagu Pada Proses Pembelajaran," *JPDI (Jurnal Pendidik. Dasar Indones.*, vol. 2, no. 2, p. 35, 2017, doi: 10.26737/jpdi.v2i2.330.
- [2] S. Susanti, S. Halidjah, and E. Uliyanti, "Pengaruh Penerapan Media Lirik Lagu terhadap Keterampilan Menulis Puisi Anak Siswa Sekolah Dasar," *J. Pendidik. dan Pembelajaran Khatulistiwa*, vol. 7, no. 11, pp. 1–9, 2018.
- [3] H. Pratiwi, I. Arfyanti, and D. Kurniawan, "Implementasi Algoritma Brute Force dalam Aplikasi Kamus Istilah Kesehatan," *J. Ilm. Teknol. Inf. Terap.*, vol. 2, no. 2, pp. 119–125, 2016.
- [4] M. Syarif, "Implementasi Algoritma String Matching Dalam Pencarian Surat Dan Ayat Dalam Al-Quran Berbasis Web," *Indones. J. Netw. Secur.*, vol. VI, no. 2, pp. 70–76, 2017.
- [5] N. Novianti, N. Marbun, M. Zarlis, and D. Hartama, "Analisis Pencarian Arti Istilah Statistika Menerapkan Algoritma Brute Force," in *Seminar Nasional Matematika dan Terapan*, 2019, vol. 1, pp. 881–886.
- [6] A. A. Nababan and M. Jannah, "Algoritma String Matching Brute Force Dan Knuth-Morris-Pratt Sebagai Search Engine Berbasis Web Pada Kamus Istilah Jaringan Komputer," *J. Mantik Penusa*, vol. 3, no. 2, pp. 87–94, 2019.
- [7] N. Afif, "Implementasi Algoritma Brute Force Dalam Perancangan Aplikasi Penelusuran Skripsi," *J. INSTEK(Informatika Sains dan Teknol.*, vol. 3, no. 1, pp. 131–140, 2018.
- [8] A. Mirza, "Pencarian Data Tiket Maintenance Menggunakan Metode Brute Force," *J. Inform.*, vol. 2, no. 3, pp. 122–126, 2017.
- [9] R. I. Ndaumanu, "Studi Komparatif Algoritma Fisher Yates dengan Brute Force pada Permainan Kartu 24," *J. Edukasi dan Penelit. Inform.*, vol. 6, no. 1, pp. 95–100, 2020, doi: 10.26418/jp.v6i1.35948.
- [10] E. P. Cindona, D. Sakethi, and F. E. Febriansyah, "Pencarian Data Pada Sistem Bank Data Nilai Mahasiswa Jurusan Ilmu Komputer Menggunakan Metode Brute Force," *J. Komputasi*, vol. 4, no. 2, pp. 76–101, 2016.
- [11] I. Gunawan, "Penggunaan Brute Force Attack Dalam Penerapannya Pada Crypt8 Dan Csa-Rainbow Tool Untuk Mencari Biss," *J. Nas. Inform. dan Teknol. Jar.*, vol. 1, no. 1, pp. 52–55, 2016, doi: 10.30743/infotekjar.v1i1.48.
- [12] A. S. Sumi, P. Purnawansyah, and L. Syafie, "Analisa Penerapan Algoritma Brute Force Dalam Pencocokan String," *Prosiding SAKTI (Seminar Ilmu Komputer dan Teknologi Informasi)*, vol. 3, no. 2, pp. 88–92, 2018, [Online]. Available: <http://e-journals.unmul.ac.id/index.php/SAKTI/article/view>

- /1836.
- [13] B. W. Santoso, F. Sundawa, and M. Azhari, "Implementasi Algoritma Brute Force Sebagai Mesin Pencari (Search Engine) Berbasis Web Pada Database," *J. Sisfotek Glob.*, vol. 6, no. 1, pp. 1–8, 2016.
 - [14] Sidik and A. Suryoprayogo, "Implementasi Algoritma Brute Force Pada Perancangan Aplikasi Kamus Bahasa Sunda Berbasis Android," *J. Inform. dan Bisnis*, vol. 5, no. 1, pp. 1–12, 2016.
 - [15] D. Abdurahman, "Rancang Bangun Aplikasi Kamus Fisika Dasar Menggunakan Algoritma String Matching Brute Force," *Infotech J.*, vol. 4, no. 1, pp. 44–46, 2018.