

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/313773155>

IMPLEMENTASI ALGORITMA BRUTE FORCE DALAM PENCARIAN DATA KATALOG BUKU PERPUSTAKAAN

Article · May 2014

CITATIONS

4

READS

1,467

1 author:



Mesran Aan

STMIK Budi Darma

39 PUBLICATIONS 50 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



String Matching [View project](#)



Determination of Education Scholarship Recipients Using Preference Selection Index [View project](#)

All content following this page was uploaded by [Mesran Aan](#) on 16 February 2017.

The user has requested enhancement of the downloaded file.

IMPLEMENTASI ALGORITMA BRUTE FORCE DALAM PENCARIAN DATA KATALOG BUKU PERPUSTAKAAN

Mesran

Dosen Tetap Program Studi Teknik Informatika STMIK Budi Darma Medan
Jl. Sisingamangaraja No.338 Simpang Limun Medan
[http:// www.inti-budidarma.com](http://www.inti-budidarma.com) // Email : mesran.s.kom.mkom@gmail.com

ABSTRAK

Perpustakaan merupakan salah satu tempat yang banyak digunakan mahasiswa untuk mencari informasi didalam memecahkan suatu masalah yang ditemui pada proses pembelajaran. Dalam pelayanannya pada perpustakaan tidak terlepas dalam satu aktifitas pencarian data. Hal ini disebabkan karena jumlah buku yang terdapat di perpustakaan sudah sangat banyak, sehingga hal ini akan memberikan masalah dalam hal pencarian data katalog buku yang terdapat pada perpustakaan. Dalam melakukan pencarian data katalog buku perpustakaan, pencocokan *string* merupakan suatu bagian dalam proses pencarian *string* (*string searching*). Data katalog buku yang dihasilkan sangat bergantung pada teknik ataupun algoritma yang dilakukan dalam pencocokan *string* tersebut. Pada penelitian ini algoritma yang digunakan dalam pencocokan *string* yaitu *Brute Force*.

Kata-kata Kunci : *String Matching*, *Brute Force*, Katalog Buku

1. Pendahuluan

Perpustakaan merupakan salah satu fasilitas yang dimiliki oleh institusi pendidikan dalam menjalankan layanannya pada bidang pendidikan. Peranan perpustakaan sangat penting dalam menyediakan buku-buku ilmiah untuk kalangan siswa dan guru dengan upaya memberikan pelayanan yang maksimal. Oleh karena itu pihak perpustakaan selalu menambah jumlah buku yang disediakan, sehingga tahun demi tahun berjumlah sangat banyak dan juga jumlah anggota perpustakaan bertambah dari tahun ke tahun. Dengan banyaknya jumlah buku maka petugas perpustakaan tentu akan mengalami kesulitan dalam hal pencarian data katalog buku yang terdapat pada perpustakaan.

Pencocokan *string* merupakan bagian penting dari sebuah proses pencarian *string* (*string searching*) dalam sebuah dokumen. Hasil dari pencarian sebuah *string* dalam dokumen tergantung dari teknik atau cara pencocokan *string* yang digunakan. Untuk mengetahui isi dokumen yang benar sesuai dengan kebutuhan informasi, diperlukan metode pencarian *string* (*string searching*) isi dokumen yang bagus. Proses pencocokan *string* (*string matching*) yang merupakan bagian utama dalam proses pencarian *string* memegang peranan penting untuk mendapatkan dokumen yang sesuai dengan kebutuhan informasi tersebut.

String Matching merupakan salah satu algoritma yang digunakan untuk mempercepat proses pencarian kata yang diinginkan. *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*. Algoritma *string matching* telah sering digunakan sebelumnya seperti contoh pada proses pencocokan *string* berdasarkan persamaan teks data yaitu *Brute Force*. Dalam hal ini,

dipilih algoritma *brute force* karena algoritma ini dapat digunakan untuk melakukan pencarian *string* atau teks. Algoritma *brute force* adalah algoritma untuk mencocokkan *pattern* dengan semua teks antara 0 dan n-m untuk menemukan keberadaan *pattern* dalam teks. Berdasarkan arah pencariannya, algoritma ini diklasifikasikan sebagai algoritma yang membaca *string* dari kiri ke kanan.

2. Tujuan Penulisan

Tujuan penulisan penelitian ini yaitu :

- Bagaimana menerapkan Algoritma *Brute Force* untuk pencarian data dalam katalog buku perpustakaan?
- Bagaimana mengimplementasikan metode pencarian *String Matching* pada aplikasi pencarian data katalog buku perpustakaan?

3. Tinjauan Pustaka

3.1. Perpustakaan

Menurut Sulisty Basuki (1991), dalam Kamus Umum Bahasa Indonesia, pustaka artinya kitab, buku. Dalam bahasa asing, ada beberapa istilah yang artinya sama dengan perpustakaan antara lain:

1. Library (bahasa Inggris)
2. Bibliotheek (bahasa Belanda)
3. Bibliotheque (bahasa Perancis)
4. Bibliotheca (bahasa Spanyol)
5. Bibliothek (bahasa Jerman)

Semua istilah itu berasal dari kata biblia dari bahasa Yunani artinya tentang buku, kitab. Mengenai pengertian dari perpustakaan dapat ditinjau dari berbagai sudut pandangan penerangan, penelitian, perencanaan, pendidikan, koleksi, gedung dan sebagainya. Penglihatan dari berbagai sudut

pandangan ini tentu saja menciptakan pengertian perpustakaan yang berbeda-beda.

Menurut Hornby, A.S. (1968) pengertian perpustakaan sebagai berikut : Perpustakaan adalah ruangan atau gedung untuk suatu koleksi buku yang disimpan untuk bacaan; buku didalam ruangan atau gedung.

Pengertian yang disebutkan di atas hanya memberi pengertian tentang perpustakaan dari suatu sudut pandang saja yaitu dari sudut gedung ataupun koleksinya.

Ada juga pendapat yang dapat memberikan pengertian yang lebih luas tentang perpustakaan misalnya pendapat yang dikemukakan oleh Wirawan (1984) yaitu : Perpustakaan adalah koleksi bahan pustaka yang disusun secara sistematis dengan tujuan untuk pengawasan, pendidikan, penelitian, management, penyembuhan, rekreasi dan sebagainya.

Pengertian perpustakaan menurut M. Sabirin Nasution adalah sebagai berikut: Perpustakaan adalah suatu unit kerja yang bertugas mengumpulkan, menyimpan, memelihara dan mengelola pemanfaatan bahan pustaka, dengan mempergunakan sistem tertentu untuk tujuan bacaan atau penelitian.

Berdasarkan pengertian-pengertian di atas dapat ditarik kesimpulan bahwa yang dimaksud dengan perpustakaan adalah suatu gedung dimana terdapat suatu unit kerja yang bertugas mengumpulkan, menyimpan, memelihara dan mengelola pemanfaatan bahan pustaka, dengan mempergunakan sistem tertentu untuk dipergunakan oleh pemakai perpustakaan sesuai dengan kebutuhan.

3.2. Katalog

Katalog berasal dari bahasa Latin *catalogus* yang berarti daftar barang atau benda yang disusun untuk tujuan tertentu. Contoh katalog dalam pengertian umum adalah *Sophie Martin Le Catalogue*, katalog penerbit. Beberapa definisi katalog menurut ilmu perpustakaan dapat disebutkan sebagai berikut :

1. Katalog berarti daftar berbagai jenis koleksi perpustakaan yang disusun menurut sistem tertentu.
2. Katalog perpustakaan merupakan suatu rekaman atau daftar bahan pustakan yang dimiliki oleh suatu perpustakaan.

Dari definisi di atas dapat disimpulkan bahwa katalog perpustakaan merupakan daftar dari koleksi perpustakaan atau berbarap perpustakaan yang disusun secara sistematis, sehingga memungkinkan pengguna perpustakaan dapat mengetahui dengan mudah koleksi apa yang dimiliki oleh perpustakaan dan dimana koleksi tersebut dapat ditemukan (Fatmi, 2011, 13).

3.3. Buku

Buku adalah kumpulan kertas atau bahan lainnya yang dijilid menjadi satu pada salah satu ujungnya dan berisi tulisan atau gambar. Setiap sisi

dari sebuah lembaran kertas pada buku disebut sebuah halaman. Seiring dengan perkembangan dalam bidang dunia informatika, kini dikenal pula istilah e-book atau buku-e (buku elektronik), yang mengandalkan komputer dan internet (jika aksesnya online)

Klasifikasi Dewey muncul pada sisi buku-buku koleksi perpustakaan. Klasifikasi dilakukan berdasarkan subjek, kecuali untuk karya umum dan fiksi. Kodenya ditulis atau dicetak ke sebuah stiker yang dilekatkan ke sisi buku atau koleksi perpustakaan tersebut. Bentuk kodenya harus lebih dari tiga digit; setelah digit ketiga akan ada sebuah tanda titik sebelum diteruskan angka berikutnya.

Ada 10 kelas utama dalam klasifikasi Dewey, yaitu sebagai berikut :

1. 000 Komputer, informasi dan referensi umum
2. 100 Filsafat dan psikologi
3. 200 Agama
4. 300 Ilmu sosial
5. 400 Bahasa
6. 500 Sains dan matematika
7. 600 Teknologi
8. 700 Kesenian dan rekreasi
9. 800 Sastra
10. 900 Sejarah dan geografi

3.4. String Matching

String Matching adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang atau teks (Riyanarto Sarno, Yeni Anistiyasari, dan Rahimi Fitri, 2012 : 11). *String Matching* dirumuskan sebagai berikut :

$$x = x[0 \dots m-1]$$

$$y = y[0 \dots n-1]$$

Dimana :

x adalah *pattern*
 m adalah panjang *pattern*
 y adalah teks
 n adalah panjang teks

Kedua *string* terdiri dari sekumpulan karakter yang disebut alfabet yang dilambangkan dengan (σ) dan mempunyai ukuran ($|\sigma|$). *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*.

Exact Matching digunakan untuk menemukan *pattern* yang berasal dari satu teks. Contoh pencarian *exact matching* adalah pencarian kata “pelajar” dalam kalimat “saya seorang pelajar” atau “saya seorang siswa”. Sistem akan memberikan hasil bahwa kalimat pertama mengandung kata “pelajar” sedangkan kalimat kedua tidak, meskipun kenyataannya pelajar dan siswa adalah kata yang bersinonim. Algoritma *exact matching* diklasifikasi menjadi tiga bagian menurut arah pencariannya, yaitu :

1. Arah pembacaan dari kiri ke kanan.
 Algoritma yang termasuk kategori ini adalah *Brute*

Force, Morris dan Pratt (yang kemudian dikembangkan oleh Knuth, Morris, dan Pratt).

2. Arah pembacaan dari kanan ke kiri.

Algoritma yang termasuk kategori ini adalah Boyer dan Moore yang kemudian dikembangkan menjadi algoritma *turbo Boyer-Moore*, *tuned Boyer-Moore*, dan *Zhu-Takaoka*.

3. Arah pencarian yang ditentukan program.

Algoritma yang termasuk kategori ini adalah algoritma *Colussi*, *Crochemore-Perrin*.

Heuristic matching adalah teknik yang digunakan untuk menghubungkan dua data terpisah ketika *exact matching* tidak mampu mengatasi karena pembatasan pada data yang tersedia (Riyanto sarno, 2012, 26). *Heuristic matching* dapat dilakukan dengan perhitungan *distance* antara *pattern* dengan teks. *Exact* dan *heuristic matching* memiliki kelemahan dalam menemukan kata yang memiliki kemiripan makna tetapi berbeda tulisan.

3.5. Algoritma Brute Force

Algoritma *brute force* adalah algoritma untuk mencocokkan *pattern* dengan semua teks antara 0 dan n-m untuk menemukan keberadaan *pattern* dalam teks (Riyanto Sarno, Yeni Anistyasari, dan Rahimi Fitri, 2012). Di dalam pencocokan string, terdapat istilah teks dan *pattern*. Teks merupakan kata yang dicari dan dicocokkan dengan *pattern*. Sedangkan *pattern* merupakan kata yang diinputkan untuk dicocokkan. Secara rinci, langkah – langkah yang dilakukan algoritma ini saat mencocokkan string adalah:

1. Algoritma *brute force* mulai mencocokkan *pattern* dari awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter pada teks yang bersesuaian, sampai salah satu kondisi berikut terpenuhi:
 - a. Karakter di *pattern* dan di teks yang dibandingkan tidak cocok.

b. Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.

3. Algoritma kemudian terus menggeser *pattern* sebesar satu ke kanan, dan mengulangi langkah ke-2 sampai *pattern* berada di ujung teks.

Algoritma *brute force* juga memiliki kelebihan dan kelemahan. Adapun kelebihan dari algoritma *brute force* yaitu :

1. Algoritma *brute force* dapat digunakan untuk memecahkan hampir sebagian besar masalah.
2. Algoritma *brute force* sederhana dan mudah dimengerti
3. Algoritma *brute force* menghasilkan algoritma yang layak untuk beberapa masalah penting seperti pencarian, pengurutan, pencocokkan string, atau perkalian matriks.
4. Algoritma *brute force* menghasilkan algoritma baku (standart) untuk tugas-tugas komputasi penjumlahan/perkalian n buah bilangan, menentukan elemen minimum atau maksimum di dalam tabel (list).

Sedangkan kelemahan dari algoritma *brute force* yaitu sebagai berikut :

1. Algoritma *brute force* jarang menghasilkan algoritma yang mangkus (manjur).
2. Beberapa algoritma *brute force* lambat, sehingga tidak dapat diterima.
3. Tidak sekonstruktif/sekreatif teknik pemecahan masalah lainnya.

Contoh penggunaan algoritma *Brute Force* untuk pencarian *pattern* dalam teks :

Teks = DATABASE MYSQL

Pattern = MYSQL

Penyelesaian :

Langkah Ke – I														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
<i>Pattern</i>	M	Y	S	Q	L									
Indeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – II														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
<i>Pattern</i>		M	Y	S	Q	L								
Indeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – III														
------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern			M	Y	S	Q	L							

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – IV														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern				M	Y	S	Q	L						

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – V														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern					M	Y	S	Q	L					

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – VI														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern						M	Y	S	Q	L				

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – VII														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern							M	Y	S	Q	L			

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – VIII														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern								M	Y	S	Q	L		

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – IX														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern									M	Y	S	Q	L	

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

tidak cocok, geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya,

Langkah Ke – X														
Teks	D	A	T	A	B	A	S	E		M	Y	S	Q	L
Pattern										M	Y	S	Q	L

Indeks 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Cocok, *pattern* ditemukan dan pencarian berhenti di indeks ke-10

c. Data Control Language (DCL)

3.6. MySQL

Database yang digunakan dalam pembuatan program ini adalah MySQL, merupakan salah satu jenis database server yang sangat terkenal kepopulerannya disebabkan MySQL menggunakan SQL (Structure Query Language) sebagai bahasa dasar untuk mengakses databasenya.

Pada MySQL, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

Dalam SQL, terdapat 3 (tiga) subbahasa, yaitu:

a. Data Defenition Language (DDL)

b. Data Manipulation Language (DML)

4. Perancangan dan Implementasi

4.1. Rancangan Database

Adapun bentuk rancangan database dapat di lihat tabel di bawah ini.

Tabel 1 Pengarang

Nama Field	Jenis	Ukuran
KdPengarang	Varchar	3
Nama	Varchar	50
Alamat	Varchar	80
Telp	Varchar	18

Tabel 2 Kategori

Nama Field	Jenis	Ukuran
KdKategori	Varchar	3
Kategori	Varchar	30

Tabel 3 Penerbit

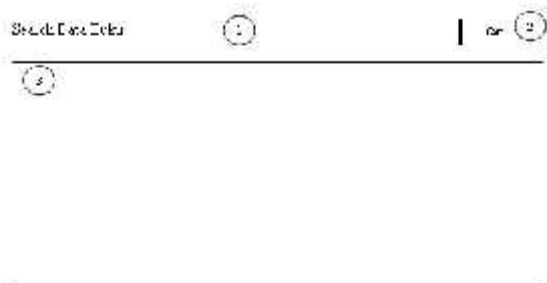
Nama Field	Jenis	Ukuran
KdPenerbit	Varchar	3
Nama	Varchar	50
Alamat	Varchar	80
Telp	Varchar	18

Tabel 4 Buku

Nama Field	Jenis	Ukuran
KdBuku	Varchar	15
Judul	Varchar	80
KdPenyusun	Varchar	3
KdPenerbit	Varchar	3
KdKategori	Varchar	3
K3=N	Varchar	20
Hal	Varchar	8
Cetakan	Varchar	3
Jumlah	Int	
Tahun	Varchar	4

4.2. Perancangan User Interface

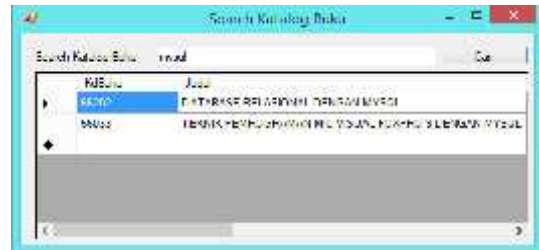
Gambar dibawah ini merupakan rancangan user interface dari pencarian data katalog perpustakaan



Gambar 3 User interface pencarian data katalog buku perpustakaan

4.3. Implementasi

Dari perancangan yang di lakukan sebelumnya dan telah di lakukan pengujian, maka sistem ini siap untuk implementasikan. Adapun hasil implementasi tersebut akan nampak pada gambar di bawah ini.



Gambar 4 Hasil implementasi algoritma brute force

5. Kesimpulan

Berdasarkan uraian di atas dapat ditarik beberapa kesimpulan, yaitu :

1. Aplikasi pencarian membantu perpustakaan dalam memberikan layanan terhadap pengunjung perpustakaan untuk melakukan pencarian terhadap katalog buku perpustakaan dalam waktu singkat.
2. Penerapan algoritma *Bruto Force* dapat melakukan pencocokan string dan memberikan hasil yang di inginkan dalam pencarian data katalog buku perpustakaan

6. Daftar Pustaka

- [1] Didik Dwi Prasetyo, 2005, "Mengolah Database dengan Visual Basic.NET dan MySQL Server", PT. Elex Media Komputindo, Jakarta
- [2] Fatmi, 2011, "Katalog Perpustakaan", Andi Yogyakarta.
- [3] Fathansyah, 2004, "Basis Data", Informatika, Cetakan I, Bandung
- [4] Nugroho, Bunafit, 2005, "Database Relasional Dengan MySQL", Andi, Jakarta
- [5] Sarno Riyanto, dkk, 2012, "Semantic Search Pencarian Berdasarkan Konten", Andi Yogyakarta.