

# Perancangan Aplikasi Kumpulan Komik Berbasis Android Menggunakan Algoritma Crochemore Perrin

Abdul Haris Siregar, Nelly Astuti Hasibuan

Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

Email: ahariss886@gmail.com

Submitted: 02/09/2020; Accepted: 27/09/2020; Published: 30/09/2020

**Abstrak**—Komik adalah sebuah karya seni yang didesain dengan menggunakan kumpulan gambar-gambar yang tidak bergerak dan dilengkapi oleh sebuah teks atau cerita. Komik merupakan sebuah buku bacaan yang di butuhkan oleh anak-anak dan juga remaja. Dalam komik, selain dari pada isi ceritanya yang menarik ada juga pesan moral yang tertanam pada setiap ceritanya. Namun semakin berkembangnya teknologi banyak sekali muncul komik-komik yang ceritanya tidak mendidik karakter remaja, mulai dari gambar serta tak adalagi pesan moral yang beredukasi yang tertanam dalam cerita tersebut. Adapun solusi yang ditarik dari permasalahan tersebut ialah dengan cara merancang suatu aplikasi komik berbasis android yang dapat diakses dan dipahami secara mudah oleh masyarakat khususnya para remaja. Namun karena keterbatasan aplikasi smartphone untuk menyajikan episode komik, maka sangat diperlukan string matching untuk pencarian judul komik. Algoritma string matching merupakan suatu algoritma pencocokan string yang sifatnya mencari sebuah string yang terdiri dari beberapa pattern (karakter) dalam jumlah teks yang besar. Untuk aplikasi komik berbasis android dirancang menggunakan algoritma crochemore-perrin.

**Kata Kunci:** String Matching, Pencarian, Komik, Crochemore Perrin, Android.

**Abstract**—Comic is a work of art that is designed using a collection of still images and is complemented by a text or story. Comics are a reading book needed by children and teenagers alike. In comics, apart from the interesting content of the stories, there are also moral messages embedded in each story. However, with the development of technology, there are many comics whose stories do not educate teenagers' characters, starting from pictures and no more educational moral messages embedded in the stories. The solution drawn from this problem is by designing an Android-based comic application that can be easily accessed and understood by the public, especially teenagers. However, due to the limitations of smartphone applications to present comic episodes, string matching is needed to search for comic titles. The string matching algorithm is a string matching algorithm that looks for a string consisting of several patterns (characters) in a large amount of text. For android-based comic applications, it is designed using the Crochemore-perrin algorithm.

**Keywords:** String Matching, Search, Comic, Crochemore Perrin, Android

## 1. PENDAHULUAN

Kamus Besar Bahasa Indonesia menjelaskan komik adalah cerita bergambar (dimajalah, surat kabar, atau berbentuk buku) yang umumnya mudah dicerna dan lucu. Komik adalah cerita yang bertekanan pada gerak dan tindakan yang ditampilkan lewat urutan gambar yang dibuat secara khas dengan paduan kata-kata. Secara umum komik adalah cerita bergambar yang ada gelembung-gelembung atau balon udara.

Bagi masyarakat khususnya para muda mudi banyak yang mengetahui makna dari setiap jenis komik yang berada didaerah mereka masing-masing. Hal itu disebabkan karena banyak media online sekarang sudah memposting komik. Akan sangat sulit jika tidak adanya pencarian dalam mencari komik yang diinginkan, mengingat saat ini sudah banyak komik terbaru.

Untuk itu perancangan aplikasi ini dibuat untuk memudahkan para muda mudi untuk mencari komik di perangkat mereka. Aplikasi yang dirancang ini merupakan sebuah aplikasi yang akan memperkenalkan macam-macam komik yang bersifat offline. Hal ini dapat membantu mempermudah bagi muda-mudi yang ingin sekali membaca komik tanpa menggunakan koneksi internet serta dapat meminimalkan biaya dalam pembelian buku komik.

Namun karena keterbatasan aplikasi smartphone untuk menyajikan episode komik, maka sangat diperlukan string matching untuk pencarian nama komik. Algoritma string matching merupakan suatu algoritma pencocokan string yang sifatnya mencari sebuah string yang terdiri dari beberapa pattern (karakter) dalam jumlah teks yang besar. Untuk aplikasi komik berbasis android dirancang menggunakan algoritma crochemore-perrin.

## 2. METODE PENELITIAN

### 2.1 String Matching

*String matching* adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang atau teks. Pattern dilambangkan dengan  $x = x[0..m-1]$  dan panjangnya adalah  $m$ . Teks dilambangkan dengan  $y = y[0..n-1]$  dan panjangnya adalah  $n$ . *String matching* dibagi menjadi dua, yaitu *exact matching* dan *heuristic matching*.

### 2.2 Algoritma Crochemore-perrin

Algoritma Crochemore-perrinn, yang sering juga disebut algoritma Two Way Algorithm, atau Algoritma Dua Arah dipublikasikan Maxime Crochemore dan Dominique Perrin pada tahun 1991. Algoritma ini memfaktorkan pattern menjadi dua bagian pattern kiri, dan patternkanan sehingga  $pattern = pattern\ kiri\ pattern\ kanan$ . Fase pencocokan pada algoritma ini terdiri dari dua bagian, pertama mencocokkan karakter patternkanan dari kiri ke kanan, lalu mencocokkan karakter patternkiri dari kanan ke kiri. Hal ini diilustrasikan pada gambar 1.

Fase inialisasi pada algoritma ini menghitung faktorisasi yang baik dari pattern atas patternkiri dan patternkanan. Jika  $(u, v)$  merupakan sebuah faktorisasi dari pattern, maka sebuah pengulangan di  $(u, v)$  adalah sebuah kata  $w$ , sehingga dua persyaratan ini terpenuhi:

1.  $w$  adalah akhiran dari  $u$  atau  $u$  adalah akhiran dari  $w$
2.  $w$  adalah awalan dari  $v$  atau  $v$  adalah awalan dari  $w$

Algoritma Crochemore-perrinn memilih faktorisasi kritis (pattern kiri, attern kanan) sehingga  $|pattern\ kiri| < per(x)$  dan  $|pattern\ kiri|$  mempunyai nilai minimal. Dengan kata lain, kata  $w$  muncul di kedua sisi dari potongan  $u$  dan  $v$  dengan emungkinan overflow di kedua sisi. Panjang dari pengulangan terkecil di  $(u, v)$  disebut periode lokal, dan dinotasikan dengan  $r(u, v)$ .

Setiap faktorisasi dari  $(u, v)$  paling tidak mempunyai satu pengulangan. Dapat dilihat dengan mudah bahwa  $1 \leq r(u, v) \leq |x|$ . Faktorisasi  $(u, v)$  dari  $x$  sehingga  $r(u, v) = per(x)$  disebut faktorisasi kritis dari  $x$ . Jika  $(u, v)$  adalah faktorisasi kritis dari  $x$ , maka pada posisi pada  $|u|$  di  $x$ , periode lokal dan periode global akan sama. [3]

### 2.3 Komik

Komik adalah sastra bergambar yang bukan hanya buku yang menampilkan visual menarik dan menjadi sebuah hiburan murahan, melainkan bentuk komunikasi visual intelektual yang memiliki kekuatan untuk menyampaikan sebuah pesan dengan bahasa yang universal, mudah dimengerti, dan selalu diingat. Indonesia memiliki begitu besar peluang dalam menghidupkan kembali komik lokal dengan komikus besar serta melahirkan komikus-komikus muda yang berpotensi untuk menceritakan Indonesia kepada dunia melalui goresan di atas kertas. Semangat dan perjuangan yang harus dilandasi pemikiran orisinal dalam berkarya, menceritakan budaya yang majemuk dan nilai-nilai budaya yang tinggi kepada dunia, dan turut melestarikan sejarah dan peninggalan pendahulu untuk selalu dikenang anak cucu akan kebesaran bangsa Indonesia di mata dunia. Harapan terbesar untuk komik Indonesia adalah dukungan dari berbagai kalangan masyarakat, pemerintah, pusat kebudayaan, komikus, sastrawan dan komunitas-komunitas komik Indonesia, bersama berjuang untuk kemajuan komik Indonesia[4].

## 3. HASIL DAN PEMBAHASAN

Pencarian komik dirancang dengan menggunakan algoritma *crochemore perrin*. Dalam algoritma pencarian *string* termasuk algoritma *crochemore perrin* terdapat teks dan *pattern*. Pada pencarian komik yang digunakan sebagai teks adalah judul komik. Panjang *pattern* harus lebih kecil dari teks. Langkah pertama dari cara kerja algoritma *crochemore perrin* dimulai dari menentukan  $pattern_{kanan}$  dan  $pattern_{kiri}$  sebagai teks acuan pencarian. Algoritma ini akan mencocokkan teks dengan *pattern* berdasarkan  $pattern_{kanan}$  dan periode, jika karakter sesuai dengan  $pattern_{kanan}$ , beri tanda sebagai periode awal, dan lanjutkan karakter selanjutnya, kemudian proses awal pencarian selanjutnya dimulai dari periode terakhir, namu jika tidak sesuai geser karakter satu persatu.

Aplikasi pencarian komik menggunakan bahasa pemrograman android dengan *platform* yang digunakan adalah *eclipse juno*. *Eclipse* merupakan sebuah IDE (*Integrated Development Enviroment*) untuk mengembangkan perangkat lunak dan dapat dijalankan disemua *platform* (*platform-independent*), *eclipse* membutuhkan android SDK (*Software Development Kit*) *windows* sebagai alat bantu dan API (*Application Programming Interface*) dalam mengembangkan aplikasi pada *platform*, hal ini dikarenakan kinerja sistem pada android mudah untuk dioperasikan dengan tampilan yang menarik.

### 3.1 Penerapan Algoritma Crochemore-perrinn

Dalam algoritma pencocokan string, teks diasumsikan berada dalam memori, sehingga bila ingin mencari string di dalam sebuah arsip, maka semua isi arsip perlu dibaca terlebih dahulu kemudian disimpan di dalam memori. Jika pattern muncul lebih dari sekali di dalam teks, maka pencarian hanya akan memberikan keluaran berupa lokasi pattern ditemukan pertama kali. Algoritma yang digunakan dalam melakukan pencocokan string (string matching) yaitu Algoritma string matching Crochemore-perrinn. Pencarian jurnal pada mobile device dengan algoritma CrochemorePerrin diharapkan dapat menyelesaikan permasalahan pencarian, dimana algoritma Crochemore-perrinn akan memudahkan pengguna mobile device untuk mencari jurnal berdasarkan judul jurnal di dalam aplikasi. Berikut ini adalah cara kerja dari algortima Crochemore-perrinn adalah sebagai berikut :

1. Langkah pertama algortima Crochemore-perrinn dimulai dengan menentukan pattern kanan dan pettern kiri, sebagai teks acuan pencarian
2. Algoritma ini akan mencocokkan teks dengan pettern berdasarkan pettern kanan dan periode :
  - a. Jika karakter sesuai dengan pettern kanan, beri tanda sebagai periode awal, dan lanjutkan ke karakter selanjutnya, kemudian proses awal pencarian selanjutnya dimulai dari periode terakhir.

b. Jika tidak sesuai gesar karakter satu persat dan kembali ke kelangkah (a)  
 Penggunaan algoritma *crochemore perrin* untuk pencarian *pattern* dalam teks, seperti pada contoh berikut ini.

**Tabel 1.** Judul

| No | Judul Komik      |
|----|------------------|
| 1  | Akatsuki Natsume |
| 2  | Vinland Saga     |

Pada contoh kasus ini Komik yang akan dicari adalah Komik “Akatsuki Natsume” pencarian dilakukan berdasarkan judul komik, misalkan dalam pencarian digunakan kata/*pattern* “Natsume” maka proses pencarian adalah sebagai berikut :

Teks : Akatsuki Natsume

*Pattern*<sub>kiri</sub> : Akatsuki

*Pattern*<sub>kanan</sub> : Natsume

Penyelesaian :

**Tabel 2.** Langkah pertama mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | A | K | A | T | S | U | K | I | N | A | T | S | U | M | E |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   |   |   |   | 1 |   |   |   |   |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   |   | N | A | T | S | U | M | E |   |   |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

**Tabel 3.** Langkah kedua mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | A | K | A | T | S | U | K | I | N | A | T | S | U | M | E |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   |   |   |   |   | 1 |   |   |   |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   |   |   | N | A | T | S | U | M | E |   |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

**Tabel 4.** Langkah ketiga mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | A | K | A | T | S | U | K | I | N | A | T | S | U | M | E |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   |   |   |   |   |   | 1 |   |   |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   |   |   |   | N | A | T | S | U | M | E |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

**Tabel 5.** Langkah keempat mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | A | K | A | T | S | U | K | I | N | A | T | S | U | M | E |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   |   |   |   |   |   |   | 1 |   |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   |   |   |   |   | N | A | T | S | U | M |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

**Tabel 6.** Langkah kelima mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | A | K | A | T | S | U | K | I | N | A | T | S | U | M | E |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   |   |   |   |   |   |   |   | 1 |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   |   |   |   |   |   | N | A | T | S | U |

*Pattern* ditemukan, karna setiap *pattern* dan teks sesuai. Pada pencarian komik “Vinland Saga” sama halnya pencarian sebelumnya. Langkah pertama mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari.

**Tabel 7.** Langkah keenam mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks           | V | I | N | L | A | N | D | S | A | G | A |
|----------------|---|---|---|---|---|---|---|---|---|---|---|
| Periode        |   |   | 1 |   |   |   |   |   |   |   |   |
| <i>Pattern</i> |   |   |   | N | A | T | S | U | M | E |   |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan

**Tabel 8.** Langkah ketujuh mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

| Teks    | V | I | N | L | A | N | D | S | A | G | A |
|---------|---|---|---|---|---|---|---|---|---|---|---|
| Periode |   |   |   | 1 |   |   |   |   |   |   |   |

|                |               |
|----------------|---------------|
| <i>Pattern</i> | N A T S U M E |
|----------------|---------------|

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

**Tabel 9.** Langkah kedelapan mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

|                |   |   |   |   |   |   |   |   |   |   |   |
|----------------|---|---|---|---|---|---|---|---|---|---|---|
| <b>Teks</b>    | V | I | N | L | A | N | D | S | A | G | A |
| Periode        |   |   |   |   |   | 1 |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   | N | A | T | S | U | M |

Karena karakter 1 pada *pattern* dengan teks tidak sesuai, maka geser *pattern* awal sebanyak satu karakter kekanan.

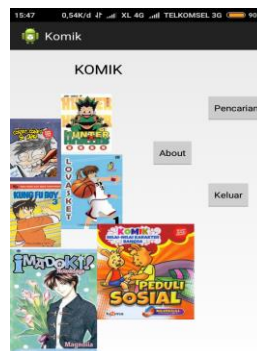
**Tabel 10.** Langkah kesembilan mencocokkan *pattern*<sub>kanan</sub> dengan teks yang akan dicari

|                |   |   |   |   |   |   |   |   |   |   |   |
|----------------|---|---|---|---|---|---|---|---|---|---|---|
| <b>Teks</b>    | V | I | N | L | A | N | D | S | A | G | A |
| Periode        |   |   |   |   |   | 1 |   |   |   |   |   |
| <i>Pattern</i> |   |   |   |   |   | N | A | T | S | U | M |

Karena karakter 1 pada *pattern* dengan teks diatas tidak memiliki kecocokan maka pencarian tidak sesuai. Maka dari kedua contoh komik diatas hanya ada 1 judul komik yang akan muncul dalam pencarian dengan *pattern* “Natsume”, yaitu “Akatsuki Natsume”

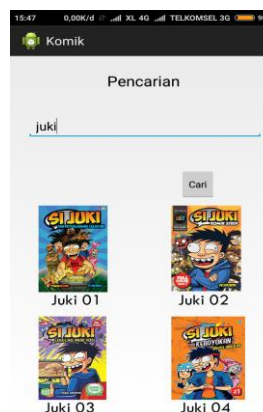
### 3.2 Implementasi

Pada Menu Utama terdapat *Button* Pencarian, About, Keluar. Tampilan menu utama dapat dilihat pada gambar berikut ini:



**Gambar 1.** Tampilan Menu Utama

Pada form pencarian user dapat memasukkan judul komik yang diinginkan seperti pada gambar dibawah ini:



**Gambar 2.** Form Menu Pencarian

Pada form hasil akan menampilkan hasil dari komik yang dicari. Tampilan dapat dilihat pada gambar 3 di bawah ini:



**Gambar 3.** Form Hasil

## 4. KESIMPULAN

Sebagai penutup pembahasan dalam penulisan skripsi ini, maka penulis mengambil kesimpulan, dimana dalam kesimpulan tersebut dapat berguna bagi pembaca. Adapun kesimpulan yang dapat ditarik adalah sebagai berikut:

1. Komik terlebih dahulu dikumpulkan dengan mencari referensi dari berbagai sumber yang ada pada internet yang kemudian disimpan kedalam sebuah database.
2. Algoritma *crochemore perrin* dapat diimplementasikan sebagai suatu solusi dalam sebuah aplikasi pencarian komik. Dimana algoritma *crochemore perrin* ini terdapat teks dan *pattern*. Panjang *pattern* harus lebih kecil dari teks.
3. Aplikasi pencarian komik dirancang menggunakan bahasa pemrograman android dengan *platform* yang dapat mengembangkan perangkat lunak dan dapat dijalankan disemua *platform (platform-independent)*.

## REFERENCES

- [1] L. Sitorus, *Algoritma dan Pemograman*. Yogyakarta: ANDI, 2015.
- [2] dkk Riyanarto, *Sematic Search Pencarian Berdasarkan Konten*. Indonesia: ANDI, 2012.
- [3] B. H. Lukmana, "PENERAPAN ALGORITMA CROCHEMORE-PERRIN PADA," vol. 17, pp. 37–42, 2018.
- [4] Tedi Sutardi, *Antropologi : Mengungkap Keragaman Budaya untu Kelas IX Sekolah Menengah Atas/Madrasah Aliyah Program Bahasa*. Bandung: PT Setia Purna Inves, 2007.
- [5] M. S. Rosa A.S, *Rekayasa Perangkat Lunak*. Bandung: informatika bandung, 2016.
- [6] Zamrony P.Juhara, *Panduan Lengkap Pemograman Android*. Yogyakarta: ANDI OFFSET, 2016.
- [7] Y. Supardi, *Pemograman Database dengan Java dan Mysql*. Yogyakarta: Elex Media Komputindo, 2010.
- [8] Mark L.Marphy, *Beginning Android*. USA, 2010.
- [9] A. A. B. Ginting and D. P. Utomo, "PERANCANGAN APLIKASI CATALOG WISATA DI SUMATERA UTARA MENGGUNAKAN ALGORITMA RABIN-KARP," *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, vol. 3, no. 1, pp. 57-63, 2019.