

PENGGUNAAN ALGORITMA PENCOCOKAN STRING UNTUK PENCARIAN NAMA PADA PHONEBOOK PERANGKAT DIGITAL

Raditya Arief

Program Studi Teknik Informatika Institut Teknologi Bandung
Jalan Ganesha 10 Bandung
e-mail: if17030@students.if.itb.ac.id

ABSTRAK

Sebuah phonebook pada perangkat digital semakin sering kita temui saat ini. Karena sudah berbentuk digital, berbagai fitur baru pun dapat ditambahkan kepada phonebook digital ini. Fitur seperti penyimpanan data hampir tidak terbatas, bertukar data secara wireless antar sesama phonebook digital, pengaturan dengan direktori, dan pencarian secara cepat menjadi nilai tambah yang hebat.

Fitur pencarian nama pun menjadi salah satu yang paling penting dalam penggunaan phonebook digital. Kecepatan dan ketepatan dalam fitur pencarian nama pada phonebook digital menjadi perhatian utama pada pembuatan fitur pencarian ini.

Dalam makalah ini penulis akan mencoba memperlihatkan bagaimana fitur pencocokan string digunakan dalam fitur pencarian nama sebuah phonebook digital.

Kata kunci: Phonebook, Pencocokan String, Brute Force, Knuth-Morris-Pratt, Boyer-Moore.

1. PENDAHULUAN

Seiring perkembangan zaman, perangkat digital pun semakin banyak penggunaannya, dan banyak perangkat digital ini yang memiliki kemampuan untuk berkomunikasi. Saat ini perangkat seperti handphone, komputer, bahkan sebuah pemutar musik digital dapat dibenamkan fitur untuk berkomunikasi.

Komunikasi adalah suatu proses penyampaian pesan, ide, atau gagasan dari suatu pihak kepada pihak lain. Cukup penting bagi kita yang sering memanfaatkan fitur komunikasi untuk menyimpan tujuan bicara kita agar tersimpan rapih dan dapat cepat ditemukan saat diperlukan. Oleh karena itu digunakanlah sebuah phonebook untuk mencatat semua tujuan komunikasi.

Phonebook, atau buku telepon dalam bahasa Indonesia, sudah merupakan suatu hal yang lumrah digunakan saat ini. Mengingat nomor telepon bukan merupakan sesuatu

yang mudah diingat setiap orang, maka penggunaan phonebook menjadi cukup penting untuk menyimpan semua nomor telepon tersebut.

Sesuai dengan perkembangan jaringan komunikasi dan cara berkomunikasi manusia, phonebook yang pada awalnya hanya digunakan untuk menyimpan nomor telepon, telah berkembang sedemikian rupa. Saat ini phonebook juga banyak digunakan untuk menyimpan alamat, email, dan lain-lain.

Karena perkembangan perangkat digital dan keragaman bentuk komunikasi yang telah dijelaskan sebelumnya, saat ini phonebook seringkali sudah terintegrasi dengan perangkat digital tersebut untuk mempermudah penggunaannya. Phonebook digital seperti ini telah dibenamkan berbagai fitur-fitur yang membuatnya lebih praktis dibandingkan dengan phonebook konvensional. Dan salah satu fitur utama paling penting yang terdapat dalam phonebook digital adalah sebuah fitur pencarian nama, karena dalam sebuah phonebook biasanya setiap nomor atau alamat kontak terbagi sesuai dengan nama tujuan yang dapat dihubungi. Dengan masukan sebuah nama dari user, maka phonebook akan menampilkan data dari kontak sesuai dengan input yang dimasukkan.

Sesuai dengan kegunaan phonebook untuk mendukung komunikasi, fitur ini harus dapat diandalkan. Pencarian yang tepat dan cepat oleh phonebook dari masukan oleh user merupakan sesuatu yang wajib pada setiap phonebook digital. Karena pencarian nama pada phonebook ini bergantung pada algoritma yang digunakannya, maka penting bagi perancang fitur ini agar memilih algoritma yang tepat untuk melakukan pencarian nama pada phonebook digital ini.

2. PHONEBOOK DIGITAL

Phonebook digital secara garis besar dapat dibagi menjadi 3 bagian. Yang pertama adalah masukan user, tipe masukan ini mutlak harus sesuai dengan tipe direktori pembagian kontak pada phonebook agar pada saat pencarian nanti tidak terjadi kesalahan. Maksudnya direktori pembagian kontak adalah bagaimana sebuah phonebook mengorganisir nomor-nomor atau data-data

yang ada dalam sebuah direktori terpisah. Dalam phonebook yang umumnya ada biasanya nomor-nomor telepon yang ada terpisah sesuai dengan nama pemilik nomor tersebut, hal ini dikarenakan satu nama dapat memiliki berbagai nomor. Namun jarang terjadi sebaliknya, bahwa satu nomor dimiliki oleh berbagai nama. Sebagai contoh, secara umum nama dalam phonebook disimpan dalam tipe String, maka masukan untuk pencarian harus juga dalam tipe String.

Bagian penting yang kedua adalah database data kontak. Data kontak ini ada berbagai macam, bergantung pada fitur yang disediakan oleh phonebook tersebut. Semua data yang disimpan ini memiliki sebuah Primary Key berupa nama direktori, nantinya apabila sebuah nama direktori dipanggil, maka semua data kontak milik nama tersebut akan ditampilkan.

Bagian penting yang terakhir adalah fitur pencarian. Dengan memasukkan sejumlah karakter dari nama kontak, user sudah dapat melihat data kontak yang terdapat dalam phonebook miliknya. Salah satu kemampuan menarik dalam fitur pencarian phonebook digital pada umumnya adalah bahwa user tidak perlu memasukkan seluruh karakter yang ada untuk menampilkan nama user yang dicari. Sebagai contoh, untuk mencari data kontak dengan nama "Ahmad Dhani", user dapat hanya memasukkan input "A", "H", dan "M", maka nama kontak Ahmad Dhani akan keluar sebagai hasilnya. Namun tentu bukan nama Ahmad Dhani saja yang akan keluar sebagai hasil apabila terdapat banyak nama kontak dengan pattern "A-H-M" pada namanya.

3. FITUR PENCARIAN PHONEBOOK

3.1 Spesifikasi Fitur

Pencarian pada phonebook berbeda dari sekadar pencocokan string biasa. Secara garis besar ada 2 hal yang membuatnya berbeda, yang pertama adalah struktur nama phonebook. Pada umumnya sebuah phonebook membagi direktori kontakannya dengan direktori nama kontak. Pada umumnya di dunia internasional, format nama yang umum diketahui mengandung Firstname, dan Lastname. Hal ini membuat user dapat mencari sebuah nama lewat Firstname maupun Lastname kontak tersebut. Sebagai contoh, untuk mencari nama Ahmad Dhani, user dapat memasukkan input "Ahmad" atau "Dhani".



Gambar 1. Interface phonebook symbian OS pada Nokia E51

Yang kedua, yaitu user dapat mencari sebuah nama hanya dengan memasukkan inputan beberapa karakter awal (pattern) dari Firstname maupun Lastname kontak. Langsung kepada contoh, untuk mencari nama Ahmad Dhani, user dapat memasukkan "A-H-M", atau "D-H-A-N". Namun satu hal yang perlu diperhatikan disini adalah, bahwa phonebook hanya memperhatikan pattern yang dimulai dari karakter awal Firstname atau Lastname. Apabila pattern tersebut terdapat di tengah string Firstname atau Lastname, maka pencarian tersebut tidak valid. Sebagai contoh untuk pencarian nama Ahmad Dhani, pencarian dengan memasukkan pattern "H-M-A" tidak akan menghasilkan keluaran Ahmad Dhani, karena pattern "H-M-A" tidak terdapat pada awalan nama kontak Ahmad Dhani.



Gambar 2. Input pattern D-E pada phonebook

3.2 Pencocokan Dengan Pattern

Menurut spesifikasi pencarian phonebook user dapat mencari nama hanya dengan pattern awal nama kontakannya, maka dari itu, algoritma yang digunakan tidak

dapat hanya menggunakan algoritma pencocokan string biasa. Untuk mencari nama kontak menggunakan pattern awal dapat digunakan algoritma pencocokan pattern (pattern matching).

4. PENCOCOKAN PATTERN

Terdapat 3 algoritma pencocokan pattern yang umum digunakan sampai saat ini, Brute Force, Knuth-Morris-Pratt (KMP), dan Boyer-Moore (BM). Ketiga algoritma ini memiliki kecepatan dan kompleksitas yang berbeda-beda. Ketiga Algoritma pencarian pattern ini juga memiliki karakteristik yang berbeda untuk berbagai jenis masalah pencarian.

4.1 Brute Force

Algoritma Brute Force melakukan pencarian secara bertahap kepada seluruh rangkaian string dari awal sampai terdapat rangkaian karakter yang cocok dengan pattern masukan user. Apabila sampai akhir string tidak juga ditemukan rangkaian karakter yang sesuai dengan pattern masukan, maka masukan dari user dianggap tidak ditemukan. Karena dimulai dari posisi awal karakter string, maka algoritma ini memiliki worst case apabila rangkaian karakter yang sesuai dengan pattern terdapat pada akhir string.

4.2 Boyer-Moore

Algoritma KMP dan BM memberikan sebuah teknik iterasi yang tidak biasa dalam pencocokan pattern pada string.

Secara umum terdapat 2 teknik yang digunakan dalam algoritma Bayer-Moore. Yang pertama adalah teknik *looking-glass*, algoritma BM memulai iterasi dari belakang pattern ke depan dimulai dari pattern paling belakang. Kemudian teknik *character-jump*, dimana saat terjadi ketidak-cocokan antara string dan pattern maka terdapat 3 cara untuk menggeser pattern.

4.3 Knuth-Morris Pratt

Sedangkan untuk algoritma KMP, pergeseran dilakukan tetap dari kiri ke kanan (seperti algoritma Brute Force). Jika pada Brute Force pergeseran dilakukan satu per satu karakter, KMP melakukan pergeseran dengan lebih pintar. Pertama-tama KMP akan mencari sebuah sub-string yang sama di awal dan akhir pattern, dan apabila terjadi ketidak-cocokan, maka pergeseran akan dibuat sedemikian rupa sehingga tidak perlu melakukan pengecekan ulang yang tidak perlu.

5. METODE PENCARIAN NAMA

Secara garis besar setiap algoritma pencocokan pattern memiliki keunggulan dan kelemahannya masing-masing. Namun dengan spesifikasi permasalahan pencarian phonebook yang sudah dijabarkan sebelumnya, penulis menyimpulkan bahwa algoritma Brute-Force adalah yang paling efektif digunakan.

Alasan penulis menetapkan Brute-Force cukup mudah, karena pada spesifikasi pencarian pattern pada phonebook hanya memperhatikan pattern yang dimulai dari awal string, sedangkan pattern yang berada di tengah string atau di akhir tidak diperhatikan. Algoritma KMP dan BM menekankan untuk melakukan sejumlah pergeseran (bukan hanya satu pergeseran) untuk menghemat proses pencocokan, sedangkan untuk pencocokan pattern pada phonebook ini tidak diperlukan pergeseran yang berarti, sehingga penggunaan KMP dan BM akan menjadi tidak efektif.

5.1 Pencocokan Nama

Hal yang perlu diperhatikan bahwa dalam phonebook tentu terdapat banyak nama yang perlu dicocokkan. Kemudian setiap nama tersebut memiliki 2 bagian yang perlu dicocokkan, yaitu Firstname, dan juga Lastname. Setelah melakukan pencocokan maka setiap nama yang memenuhi persyaratan kecocokan dengan pattern input harus ditampilkan pada interface.

Pseudo-code:

```
Procedure ceknama(input database: database, input
pattern : string)
{mencari nama dalam database yang valid dengan pattern
masukan dari user
Masukan: database dan pattern
Keluaran: nama kontak yang sesuai dengan pattern}
```

Deklarasi

```
string fname <= database.name.firstname
string lname <= database.name.lastname
{ 'name' akan diisi dengan
nama pertama dalam
daftar database }
char cfname <= firstname.firstchar
char clname <= lastname.firstchar
cpattern <= pattern.firstchar
```

Algoritma

```
{selama belum semua nama pada database diperiksa}
while(database.nextname != NULL) do
```

```

while( (cfname = cpattern or clname = cpattern) and
cpattern ≠ pattern.lastchar) do

    {mengambil karakter selanjutnya}
    cfname ≤= firstname.nextchar
    clname ≤= lastname.nextchar
    cpattern ≤= pattern.nextchar
endwhile

{apabila pengujian pattern berhasil
karakter pengujian pattern akan
sampai pada karakter pattern terakhir}

if(cpattern == pattern.lastchar)
{nama kontak akan ditampilkan}
name.show
endif

{inisiasi untuk iterasi berikutnya}
name = database.nextname
cfname = firstname.firstchar
clname = lastname.firstchar
cpattern = pattern.firstchar

endwhile

```

6. KESIMPULAN

Dalam sebuah phonebook digital memang tidak mutlak terdapat algoritma pattern matching. Namun penggunaan algoritma ini sebagai pendukung fitur phonebook dapat memberikan kemudahan kepada user karena user tidak harus mengetik seluruh nama kontak yang ingin dicari. Dari ketiga algoritma yang ada, brute force dinilai paling tepat digunakan pada masalah ini karena pada masalah ini tidak dibutuhkan pergeseran pattern yang banyak.

REFERENSI

Munir, Rinaldi. Diktat Kuliah IF2251 Strategi Algoritmik. Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung. 2006.

- [1] Munir, Rinaldi, "Diktat Kuliah IF3051 Strategi Algoritma", 2007.
- [2] www.gsmarena.com.