

## ALGORITHMMA PENGAKSESAN BASIS DATA

**Dwi Sakethi**

Jurusan Matematika FMIPA Universitas Lampung

Jl. S. Brojonegoro No. 1 Bandar Lampung 35145

*Email: dwijim@unila.ac.id*

### ABSTRACT

In information system development, the implementation of program is one step of system development. To get the best execution time, effective and efficient algorithm is required. That algorithm will significantly influence execution time of the accessing database when the amount of the data is large. This paper discusses several algorithms of accessing database by using xBase Application.

**Keywords:** *database, algorithm, xBase application*

### PENDAHULUAN

Iwan Syarif menyatakan : Algorithma berasal dari kata algoris dan ritmis, yang pertama kali diungkapkan oleh Abu Ja'far Mohammed Ibn Musa al Khowarismi (825 M) dalam buku Al-Jabr Wa-al Muqabla. Dalam bidang pemrograman algorithma didefinisikan sebagai suatu metode khusus yang tepat dan terdiri dari serangkaian langkah yang terstruktur dan dituliskan secara sistematis yang dikerjakan untuk menyelesaikan suatu masalah dengan bantuan komputer<sup>1</sup>.

Dalam pengembangan sistem informasi/perangkat lunak ada empat langkah penting yang selalu dilakukan yaitu : (1) Pendataan kebutuhan sistem, analisis, dan spesifikasi (2) Desain sistem (3) Implementasi program dan (4) Pengujian<sup>2</sup>. Salah satu tahapan dalam pengembangan sistem informasi adalah implementasi yang dalam hal ini dilakukan proses pembuatan program (*coding*) serta basis data sesuai dengan rancangan dan analisis yang telah

dilakukan. Pada tahap pembuatan program (*coding*) ini, fokus pekerjaan ditujukan kepada bagaimana menyelesaikan masalah yang ada. Jarang ada pemikiran bagaimana menyelesaikan permasalahan dengan optimal, dalam hal ini waktu yang paling cepat. Pada tahap ini, jika hasil sudah diperoleh maka biasanya pekerjaan kemudian dilanjutkan pada kegiatan berikutnya. Selain itu juga biasanya pada tahap pengembangan atau tahap awal operasional sistem, basis data yang ada belum mempunyai data dalam jumlah yang besar. Hal ini mengakibatkan algoritma yang dipakai dalam penyelesaian masalah tidak terlalu memperlambat sistem. Akan sangat berbeda jika basis data sudah memiliki data dalam jumlah yang besar.

Pada penelitian ini, akan ditinjau pengaruh algoritma yang kemudian diimplementasikan dalam bentuk program terhadap kecepatan eksekusi program. Program yang dijadikan untuk contoh adalah program untuk mencetak sejumlah data mahasiswa.

## Dasar Teori

Pemrograman Clipper termasuk salah satu sistem xBase yang pada saat ini masih digunakan baik di dunia industri maupun dunia pendidikan. Untuk melakukan akses ke basis data, perintah yang digunakan adalah **perintah use**.

Sintaks: use nama\_file\_data

Contoh: use mhsx

Dalam pemrosesan basis data, dapat digunakan *file index*. Penggunaan *file index* ini sifatnya pilihan dan tergantung model akses yang akan diimplementasikan. Penggunaan *file index* yang tepat akan mempercepat proses akses basis data.

Sintaks: index on [atribut] to [nama file]

Contoh: use mhsx

index on NPM to hasil

Untuk mengolah data menjadi suatu informasi, tidak semua data akan diproses. Pemilahan data yang akan diproses dapat dilakukan dengan berbagai macam algorithma yang kemudian diterjemahkan ke dalam perintah-perintah dalam program. Salah satunya adalah dengan melakukan penyaringan (seleksi) yang dilakukan dengan **perintah set filter to**. Penggunaannya dapat dilihat seperti berikut :

Sintaks : set filter to [kondisi]

Contoh: set filter to

NPM>='0117031001' .and. NPM <='0117031027'

Algorithma lain yang dapat digunakan untuk melakukan pemilahan data adalah dengan menggunakan indeks dan kemudian mengarahkan *pointer* ke data yang memenuhi kriteria pemilahan, memprosesnya berulang-ulang sampai ditemukan data yang tidak memenuhi

kriteria pemilahan data. Pengarahan *pointer* dilakukan dengan **perintah seek** setelah terlebih dahulu dilakukan proses indeks.

Sintaks : seek [nilai]

Contoh : seek '0117031001'

Kemudian proses *looping* dilakukan dengan **perintah do while** seperti berikut :

Sintaks : do while [kondisi]

    enddo

Contoh : do while NPM<='0117031001'

    ...

    enddo

Pencetakan data dilakukan dengan **perintah say** yang penulisannya memiliki aturan seperti berikut:

Sintaks : @ baris,kolom say [suatu\_nilai]

Contoh : @ prow()+1,0 say npm+'  
'+nama

Untuk mengetahui lamanya waktu yang digunakan oleh proses akses basis data ini, pada awal proses dicatat jam mulai dan pada akhir proses dicatat jam selesai. Dengan demikian lama waktu dihitung dengan mencari selisih waktu awal dan waktu selesai. Pencatatan waktu dilakukan dengan **perintah seconds()**. Dengan contoh seperti berikut :

awal = seconds()

...

akhir = seconds()

@ 4,1 say akhir-awal

## METODE PENELITIAN

Alat yang digunakan dalam penelitian ini adalah pemrograman xBase dengan menggunakan Clipper 5.2 yang dijalankan pada komputer Compaq Pentium III 850 MHz RAM 128 MB menggunakan *Command Prompt Only* yang diambil dari sistem operasi Windows 98<sup>3,4</sup>. Data yang digunakan berupa

contoh data mahasiswa yang terdiri dari atribut NPM dan Nama Mahasiswa dengan jumlah data sebanyak 35190 *record*. Dalam proses ini dari data contoh tersebut akan dicetak data mahasiswa Matematika sebanyak 27 orang dengan NPM mulai dari 0117031001 sampai dengan 0117031027. Proses pengaksesan dilakukan dengan tiga model algoritma yaitu:

1. Pengarahan *pointer* ke *record* 0117031001 menggunakan **perintah seek** dan **perintah index**. Pemrosesan dilakukan secara berurutan sampai dengan NPM 0117031027. Hal ini dimungkinkan karena data sudah diindeks berdasarkan NPM. Algoritma ini diimplementasikan dalam program **sel.exe**.
2. Penggunaan indeks yang digabung dengan pemilahan data menggunakan **perintah set filter to**. Konsekuensinya pemrosesan dilakukan dari *record* pertama sampai dengan *record* terakhir. Namun karena *set filter to* dilakukan dengan perintah *set filter to NPM >= '0117031001' .and. NPM <= '0117031027'* maka data yang diproses hanya NPM dari 0117031001 sampai dengan 0117031027. Algoritma ini diimplementasikan dalam program **sele.exe**.
3. Penggunaan **perintah set filter to** tanpa menggunakan indeks. Algoritma ini sama dengan model pada nomor 2 tersebut di atas tapi tanpa menggunakan indeks. Algoritma ini diimplementasikan dalam program **select.exe**.

### HASIL DAN PEMBAHASAN

Dari hasil eksekusi program dengan aplikasi xBase<sup>3,4</sup>, maka diperoleh waktu eksekusi sebagai berikut :

**Tabel 1. Waktu eksekusi**

No	sel.exe	sele.exe	select.exe
1	0,27	17,25	8,02
2	0,33	16,80	8,13
3	0,33	17,08	7,91
4	0,27	16,59	7,75
5	0,28	16,80	7,96
6	0,27	16,59	7,58
7	0,33	16,86	7,74
8	0,33	16,86	7,41
9	0,33	16,75	7,42
10	0,33	16,48	7,52
11	0,33	16,70	7,69
12	0,38	16,86	8,30
13	0,27	16,70	7,85
14	0,33	16,42	7,96
15	0,33	16,81	8,12
16	0,28	16,81	8,02
17	0,33	16,75	8,02
18	0,39	16,48	7,97
19	0,33	16,86	8,02
20	0,33	16,48	7,52
21	0,33	16,48	7,58
22	0,33	16,75	7,74
23	0,38	16,31	7,41
24	0,32	16,81	8,02
25	0,38	16,59	7,58
26	0,33	16,70	7,53
27	0,33	16,31	7,58
28	0,33	16,15	7,53
29	0,33	16,26	7,47
30	0,33	16,54	7,41
31	0,33	16,26	7,58
32	0,33	16,81	7,53
33	0,33	16,04	7,52
34	0,39	16,75	7,58
35	0,33	16,37	7,47

Selanjutnya diperoleh nilai rata-rata, standar deviasi, maksimum dan minimum waktu eksekusi sebagai berikut seperti terdapat pada Tabel 2 di bawah ini:

**Tabel 2. Nilai rata-rata, standar deviasi, maksimum dan minimum waktu eksekusi**

sel.exe	sele.exe	select.exe
Rata-rata :		
0,3277	16,6303	7,7269
Standar Deviasi :		
0,0316	0,2618	0,2541
Maksimum :		
0,3900	17,2500	8,3000
Minimum :		
0,2700	16,0400	7,4100

### KESIMPULAN

Dari hasil pengukuran waktu yang digunakan untuk proses eksekusi program terlihat bahwa algorithma akses basis data dengan menggunakan indeks dan pengarahan *pointer* ke data yang diperlukan dan kemudian memprosesnya hanya sampai dengan data yang

memenuhi kriteria lebih efisien. Dari sini dapat disimpulkan juga perlunya penggunaan algorithma yang efisien dalam pengaksesan basis data supaya proses berjalan dengan cepat tanpa mengabaikan kebutuhan informasi yang diperlukan.

### DAFTAR PUSTAKA

1. Budiyanto, A., 2003. Pengantar Algorithma dan Pemrograman, [www.ilmukomputer.com](http://www.ilmukomputer.com),
2. Harvey Mudd College Computer Science Departement, 2005. [www.cs.hmc.edu](http://www.cs.hmc.edu).
3. Pearson, D., 2002. The Guide To Clipper 5.2, [www.clipx.net](http://www.clipx.net).
4. Sakethi, D., 2003. Pemrograman Clipper, Ilmukomputer.com.