

ALGORITMA SEARCHING

Dalam bentuknya yang paling dasar, masalah searching dinyatakan sbb.:

Jika ada satu deret $S = \{s_1, s_2, \dots, s_n\}$ yg terdiri dari integer dan ada integer x , diminta untuk menentukan apakah $x = s_k$ untuk s_k yg ada di S .

procedure SEQUENTIAL SEARCH (S, x, k)

Step 1: (1.1) $i \leftarrow 1$

(1.2) $k \leftarrow 0$

Step 2: while ($i \leq n$ and $k = 0$) do

 if $s_i = x$ then $k \leftarrow i$ end if

$i \leftarrow i+1$

end while

Worst case: waktu = $O(n)$

SEARCHING DERET TERURUT

$S = \{s_1, s_2, \dots, s_n\}$ terurut naik, yaitu, $s_1 \leq s_2 \leq \dots \leq s_n$.

Yang di-search adalah file dgn n record dan terurut berdasar field s .

s_i	INFORMASI LAIN
-------	----------------

EREW (Exclusive Read Exclusive Write) SEARCHING

- Tersedia N -prosesor komputer EREW SM SIMD untuk search S untuk elemen x , dgn $1 < N \leq n$.
- Nilai x harus diketahui semua prosesor. Dipakai prosedur BROADCAST dlm waktu $O(\log N)$.
- Deret S dibagi menjadi N subsequence, masing-masing dgn panjang n/N .
- Semua prosesor melakukan prosedur BINARY SEARCH yg membutuhkan waktu $O(\log(n/N))$ worst case.
- Elemen S dianggap berbeda semua, shg paling banyak satu prosesor menemukan s_k sama dgn x dan mengembalikan k .
- Waktu total: $O(\log N) + O(\log(n/N)) = O(\log n) \rightarrow$ sama dengan BINARY SEARCH yg berjalan pd satu prosesor, dgn demikian tdk ada penambahan kecepatan.

CREW (Concurrent Read Exclusive Write) SEARCHING

- Tersedia N -prosesor komputer CREW SM SIMD untuk search S untuk elemen x , dgn $1 < N \leq n$.
- Algoritma sama dgn EREW, kecuali semua prosesor membaca x bersamaan dlm waktu konstan dan kemudian melakukan prosedur BINARY SEARCH pd subsequence masing-masing.
- Waktu: $O(\log(n/N))$ worst case \rightarrow lebih cepat dari BINARY SEARCH yg berjalan pd satu prosesor.

SEARCHING DERET ACAK

Deret $S = \{s_1, s_2, \dots, s_n\}$ tdk urut dan belum tentu memiliki elemen yg berbeda semua.

Search spt ini dinamakan juga *query*.

Search juga berguna untuk *maintenance* file, misalnya menyisipkan (insert) record baru dan men-delete record yg ada.

Searching pd komputer SM SIMD

Tersedia N -prosesor komputer untuk search S untuk elemen x , dgn $1 < N \leq n$.

procedure SM SEARCH (S, x, k)

Step 1: **for** $i = 1$ to N **do in parallel**

Read x

end for

Step 2: **for** $i = 1$ to N **do in parallel**

(2.1) $S_i \leftarrow \{s_{(i-1)(n/N)+1}, s_{(i-1)(n/N)+2}, \dots, s_{i(n/N)},$

(2.2) SEQUENTIAL SEARCH (S_i, x, k_i)

end for.

Step 3: **for** $i = 1$ to N **do in parallel**

if $k_i > 0$ then $k \leftarrow k_i$ end if

end for.

Analisis:

EREW:

- Step 1 : diimplemntasi dgn menggunakan prosedur BROADCAST dan memerlukan waktu $O(\log N)$.
- Step 2 : prosedur SEQUENTIAL SEARCH memerlukan $O(n/N)$ worst case.
- Step 3 : prosedur STORE berjalan dlm waktu $O(\log N)$.

- Running time asimptotik total:

$$t(n) = O(\log N) + O(n/N),$$

dan biaya sebesar:

$$c(n) = O(N \log N) + O(n) \text{ yg tdk optimal.}$$

ERCW:

Step 1 dan 2 sama dgn kasus EREW, sementara step 3 membutuhkan waktu konstan. Running time asimptotik total tetap tdk berubah.

CREW:

Step 1 sekarang membutuhkan waktu konstan, sementara step 2 dan 3 sama dgn kasus EREW. Running time asimptotik total tetap tdk berubah.

CRCW:

Kedua langkah 1 dan 3 membutuhkan waktu konstan, sementara step 3 sama dgn kasus EREW. Running time total sekarang sebesar $O(n/N)$, dan biaya sebesar:

$$c(n) = N \times O(n/N) = O(n)$$

yg optimal.