

Sistem

Operasi

ALIF ALFABRI.B (12050112600) - TIFAC

1. diketahui sistem memori demand paging. page tabel menggunakan register. membutuhkan 8 millisecond untuk mengganti page fault jika frame kosong terse dia atau page yg direplace tidak dimodifikasi dan 20 millisecond jika page yg direplace dimodifikasi. waktu akses memori adalah 100 nanosecond. diasumsikan page yg direplace akan dimodifikasi adalah 70 persen dari waktu. berapa rata-rata? page fault yg diterima untuk effective access time tidak lebih dari 200 nanosecond?

$$Lb\ Ma = 100\ ns$$

$$EAT = 200\ ns$$

$$Page_Fault_time\ (Oframe) = 8\ ms = 8.000.000\ ns$$

$$Page_Fault_time\ (with\ frame) = 20\ ms = 20.000.000\ ns$$

$$\begin{aligned} EAT\ (Oframe) &= (1 - p_1) \times Ma + p_1 \times Page_Fault_time \\ &= (1 - p_1) \times 100 + p_1 \times 8.000.000 = 100 - 100p_1 + 8.000.000p_1 \\ 200 &= 100 + 7.999.900p_1 \\ 100 &= 7.999.900p_1 \\ p_1 &= 0,0000125 \end{aligned}$$

$$\begin{aligned} EAT\ (with\ frame) &= (1 - p_2) \times Ma + p_2 \times Page_Fault_time \\ &= (1 - p_2) \times 100 + p_2 \times 20.000.000 \\ 200 &= 100 + 19.999.900p_2 \\ 100 &= 19.999.900p_2 \\ p_2 &= 0,0000005 \end{aligned}$$

$$\begin{aligned} rata-rata\ page\ fault &= p_1 \times 30\% + p_2 \times 70\% \\ &= 0,000000375 + 0,00000035 \\ &= 0,00000073 \end{aligned}$$

2. diketahui string awal page:

1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6

berapa banyak page fault yg terjadi untuk algoritma page replacement berikut de Ugan satu, dua tiga, empat, lima, enam / 7 frame? ingat bahwa semua frame di inisialisasikan kosong, sehingga setiap page unik pertama akan bernilai satu sing - masing satu fault.

a. LRU

b. FIFO

c. Optional

Lb a. halaman 7 kesalahan : 1 frame : 7 - 1 - 1 waktu : 1

0	2	7	0	-1	1	2	b. page 7 faults: 1 frames: 7 - 1 - 1 time: 1
1	3	7	0	1	1	2	0 2 7 0 -1 1 2
2	4	2	0	1	4	2	1 3 7 0 1 1 2 3
0	4	2	0	1	4	5	2 4 2 0 1 4 2 3
3	5	2	0	3	4	5	0 4 2 0 1 4 2 3
0	5	2	0	3	4	7	3 5 2 3 1 4 6 3
2	7	4	0	2	8	7	0 6 2 3 0 4 6 7
3	8	4	3	2	8	10	4 7 4 3 0 8 6 7
0	9	0	3	2	11	10	2 8 4 2 0 8 9 7
3	9	0	3	2	11	12	3 9 4 2 3 11 9 10
2	9	0	3	2	11	12	0 10 0 2 3 11 9 10
1	10	1	3	2	14	12	2 10 0 2 3 11 9 10
2	10	1	3	2	14	12	1 11 0 1 3 11 14 10
0	11	1	0	2	14	16	2 12 0 1 2 11 14 15
1	11	1	0	2	17	16	0 12 0 1 2 11 14 15
7	12	1	0	7	17	16	1 12 0 1 2 11 14 15
0	12	1	0	7	17	19	7 13 7 1 2 18 14 15
1	12	1	0	7	20	19	0 14 7 0 2 18 19 15
							1 15 7 0 2 18 19 20

c. Page : 7 faults: 1 frames: 7 - 1 - 1

0	2	7	0	-1	3	7	2	0	3
1	3	7	0	1	2	7	2	0	3
2	4	2	0	1	1	8	2	0	1
0	4	2	0	1	2	8	2	0	1
3	5	2	0	3	0	8	2	0	1
0	5	2	0	3	1	8	2	0	1
4	6	2	4	3	7	9	7	0	1
2	6	2	4	3	0	9	7	0	1
3	6	2	4	3	1	9	7	0	1
0	7	2	0	3					

3. diketahui array 2 dimensi A sebagai berikut : Var A : array [1.....100] of integer.

Diketahui A [1][1] pada lokasi 200 pada sistem page memori dengan page - page berukuran 200. Suatu proses kecil pada page 0 (lokasi 0 s/d 100) untuk manipulasi matriks, selanjutnya setiap matriks dimulai dari page 0. untuk 3 frame page berapa banyak page fault yg disebabkan oleh loop inisialisasi array berikut menggunakan loop dan asumsi frame page 1 sudah terdapat proses dan 2 frame page lainnya diinisialisasi kosong.

a. for (j = 1 ; j <= 100 ; j++)


```
for ( i = 1 ; i <= 100 ; i ++ )
```

```
    A [ i ] [ j ] = 0 ;
```

↳ dengan ukuran halaman 200, setiap halaman dapat menampung 200 nilai. Maka iterasi loop adalah untuk salin. Ketika $A[i][j]$ diakses dengan cara ini, di iterasi pertama inner loop, $A[0][0]$ hingga $A[i][99]$ akan diambil dari satu blok dan dalam pengulangan ke-2 dari loop dalam, $A[2][0]$ ke $A[3][99]$ akan diambil dalam blok yang tersedia kedua dan ini berlanjut. Kesalahan halaman dimatikan untuk setiap iterasi alternatif dari loop dalam. Jadi total 50 halaman kesalahan untuk loop dalam loop luar akan menyebabkan kesalahan halaman untuk setiap iterasi outer. Jadi, jumlah total page fault = $50 \times 100 = 5000$.

```
b) for ( i = 1 ; i <= 100 ; i ++ )
```

```
    for ( j = 1 ; j <= 100 ; j ++ )
```

```
        A [ i ] [ j ] = 0 ;
```

↳ 50 disini, karena loop dalam adalah untuk j, 200 entri yang diambil dalam halaman kesalahan akan diakses secara berurutan dan kemudian menghasilkan kesalahan halaman lain untuk mengambil 200.000 penempatan. Yang itu $A[0][0]$ hingga $A[i][99]$ yang diambil di halaman pertama akan diurutkan dalam dua iterasi pertama dari loop luar. Iterasi ketiga loop luar akan menghasilkan kesalahan halaman kedua dan seterusnya. Kesalahan halaman akan dihasilkan untuk setiap baris alternatif $A[i][j]$ ($A[0][0]$, $A[2][0]$, $A[4][0]$, ..., $A[98][0]$). Jadi jumlah total page fault adalah $100/2 = 50$.

4. diketahui sistem demand paging dengan paging disk mempunyai waktu akses dan transfer rata-rata 20 msec. diambil ditranslasikan melalui page tabel di

memori, dengan waktu akses 1 micro sec per akses memori. sehingga akan ke memori melalui page tabel sama dengan 2 kali akses memori untuk memperbarui waktu, ditambahkan associative memory yg menurunkan waktu akses menjadi satu akses memori. jika entri page table berada di associative memory. diasumsikan 80% akses pada associative memory dan dari sisanya (20%), 10% nya (atau 2 persen dari total) menyebabkan page fault. berapakah effective access time-nya

↳ diketahui : waktu akses 20 mS rata-rata transfer 20 mS, memory access = 400 mS

$$\begin{aligned} EAT &= (1-p) \times (400) + p \times (20mS) \\ &= (1-p) \times (400) + p \times 20000000 \\ &= 400 + 19999600 \times p \end{aligned}$$

$$410 > 400 + 20.000.000p$$

$$10 > 20000000p$$

$$p < 0,0000005$$