

Tugas 5.11

1. M<sup>b</sup>alkan sebuah algoritma pengadukan mengutamakan proses<sup>2</sup> yg telah menggunakan waktu prosesor paling sedikit dlm waktu dulu. Mengapa algoritma ini akan menguntungkan program<sup>2</sup> yg bergantung pd I/O dan tidak membuat program<sup>2</sup> yg bergantung pd CPU kelaparan secara permanen?

Algoritma pengadukan yg memprioritaskan proses<sup>2</sup> dg penggunaan waktu prosesor paling sedikit akan menguntungkan program I/O karena program ini hanya menggunakan CPU sebentar, sebelum menunggu I/O lagi. Karena mereka sering memiliki waktu prosesor yg rendah, mereka lebih sering tergantung.

Tetapi program yg bergantung pd CPU, menbutuhkan waktu pemrosesan lbh banyak, tdk ada kelaparan karena mereka tdk digantung, waktu prosesor mereka mampu. Sering waktu, ini membuat mereka lebih menguntungkan dan algoritma. Jadi algoritma ini memastikan bahwa program I/O mendapatkan waktu CPU yg mereka butuhkan, sementara program CPU juga mendapatkan bagiannya dan tdk kelaparan permanen.

2. M<sup>b</sup>alkan anda memiliki pekerjaan<sup>2</sup> berikut yg harus dieksekusi dg satu prosesor, dg pekerjaan<sup>2</sup> datang dlm urutan yg terdapat di bawah ini:

i	T (Pi)
0	80
1	20
2	10
3	20
4	50

a) M<sup>b</sup>alkan sebuah sistem menggunakan penjatuhan FCFS. Buatlah diagram GANTT yg menggambarkan eksekusi proses<sup>2</sup> ini!

P0	P1	P2	P3	P4
----	----	----	----	----

0 80 100 110 130 180

b. Berapa waktu penyelesaian utk proses P3

Completion time - arrival time

$$= 130 - 0 = 130$$

c. Berapa waktu rata-rata utk proses2 ini?

$$(p_1 + p_2 + p_3 + p_4) / 5$$

$$= (8 + 80 + 100 + 110 + 130) / 5$$

$$= 420 / 5 = 84$$

3

M<sup>2</sup>Balkan sebuah proses baru dlm sebuah sistem tiba dg rata-rat<sup>2</sup> empat proses per menit dan setiap proses tsb memerlukan waktu pelayanan 8 detik. Perkirakan fraksi waktu CPU sibuk dlm sebuah sistem dg satu prosesor!

Jumlah proses per menit = 6

Rata-rat<sup>2</sup> waktu pelayanan per proses = 8 s

Total waktu pelayanan per menit =  $6 \times 8 = 48$  s

Faksi waktu CPU sibuk =  $48 / 60 = 0,8 = 80\%$

4.

Sebuah algoritma pengjadwalan CPU memerlukan urutan utk eksekusi proses<sup>2</sup> yg dijadwalkan. Diberikan n proses yg harus dijadwalkan pd satu prosesor, berapa banyak kemungkinan jadual yg berbeda? Berikan sebuah rumus dlm bentuk n!

Untuk menentukan jumlah kemungkinan jadual yg berbeda utk n proses, kita menggunakan jumlah permutasi dari n elemen.

$$n! = n \times (n-1) \times (n-2) \times \dots \times 1 \quad nPr = \frac{n!}{(n-r)!}$$

M<sup>2</sup>Balk ada 3 proses, maka:

$$\begin{aligned} 3P_1 &= \frac{3!}{(3-1)!} \\ &= \frac{3 \times 2 \times 1}{2} = 3 \end{aligned}$$

5-

Bawlah algoritma pengjadwalan CPU yg diparameterisasi. Contoh, algoritma RR memerlukan parameter utk manajemen size makin. Antaraan wmpm bslk multilevel memerlukan parameter utk mendefinisikan

Jumlah antren, algoritma pengaduhan untuk setiap antren ketika yg digunakan untuk memudahkan proses antren antren, dan sebagainya.

Algoritma ini sebenarnya adalah kumpulan algoritma fungsi, kumpulan algoritma RR (dengan sentra slice waktu), dan sebagainya. Serta kumpulan algoritma mungkin mencakup yg lain (misalnya, algoritma FCFS adalah algoritma RR dg kuantum waktu tak terbatas). Apa hubungan (jika ada) antara pasangan set algoritma berikut?

#### a. Prioritas dan SJF

SJF (Shortest Job First) adalah bentuk khusus dari algoritma prioritas diberikan berdasarkan panjang burst waktu (CPU (sebaliknya pendek burst, semakin tinggi prioritasnya)).

#### b. Antren umpan balik multilevel dan FCFS

Algoritma antren umpan balik multilevel bisa mencakup FCFS jika salah satu antren menggunakan pendekatan FCFS.

#### c. Prioritas dan FCFS

Algoritma prioritas bisa memiliki pertolongan yg sama dg FCFS jika semua proses memiliki prioritas yg sama.

#### d. RR dan SJF

Tidak ada hubungan langsung dari RR (Round Robin) dan SJF (Shortest Job First) karena keduaanya menggunakan pendekatan pengaduhan yg berbeda.

### 6. Bandingkan antara pengaduhan jangka panjang dan pengaduhan jangka pendek!

Pengaduhan jangka panjang

Pengaduhan jangka pendek

- Juga dikenal dg pengaduhan tugas

- Dikenal dg pengaduhan CPU

- Frequnensi jarang terjadi

- Frequnensi tinggi (setiap milisekon)

- Tujuannya untuk menentukan program mana yg akan dijalankan masuk ke dalam sistem dg direksa

- Tujuannya adalah menentukan proses mana dalam antren stop yg akan direksa selanjutnya pd CPU

- Bergantung mengontrol tingkat multiprogramming - Bergantung memastikan (PV ejeksi

7. Perombongkan rangkaian proses berikut ini, dg penging burst CPU yg diberikan dimulidik!

Process	Burst time	Priority
P <sub>1</sub>	10	3
P <sub>2</sub>	4	1
P <sub>3</sub>	2	3

Process	Burst time	Priority
P <sub>4</sub>	1	4
P <sub>5</sub>	5	2

- a. Buat empat diagram garis yg menggambarkan eksekusi proses2 ini menggunakan algoritma pagiduan berikut: FCFS, SJF, prioritas non-preemptive (angka prioritas yg lebih kecil menunjukkan prioritas tsb tinggi), dan RR (quantum=1)!

FCFS | P<sub>1</sub> | P<sub>2</sub> | P<sub>3</sub> | P<sub>4</sub> | P<sub>5</sub> |

0 10 11 13 14 19

SJF | P<sub>2</sub> | P<sub>4</sub> | P<sub>3</sub> | P<sub>5</sub> | P<sub>1</sub> |

0 1 2 4 9 19

Prioritas non-preemptive | P<sub>2</sub> | P<sub>5</sub> | P<sub>1</sub> | P<sub>3</sub> | P<sub>4</sub> |

0 1 6 16 18 19

RR (Round Robin; Quantum=1)

| P<sub>1</sub> | P<sub>2</sub> | P<sub>3</sub> | P<sub>4</sub> | P<sub>5</sub> | P<sub>1</sub> | P<sub>3</sub> | P<sub>5</sub> | P<sub>1</sub> | P<sub>5</sub> | P<sub>1</sub> |

0 1 2 3 4 5 6 7 8 9 10 11 12

- b. Berapa waktu turnaround utk setiap proses utk masing2 algoritma pagiduan pd bagian (a)?

FCFS P<sub>1</sub>=10 P<sub>2</sub>=11 P<sub>3</sub>=13 P<sub>4</sub>=14 P<sub>5</sub>=19

SJF P<sub>2</sub>=1 P<sub>4</sub>=2 P<sub>3</sub>=4 P<sub>5</sub>=9 P<sub>1</sub>=19

Prioritas Non-Preemptive P<sub>2</sub>=1 P<sub>5</sub>=6 P<sub>1</sub>=16 P<sub>3</sub>=18 P<sub>4</sub>=19

RR P<sub>1</sub>=19 P<sub>2</sub>=2 P<sub>3</sub>=7 P<sub>4</sub>=4 P<sub>5</sub>=3

- c. Berapa waktu tunggu utk setiap proses utk masing2 algoritma pagiduan pd bagian (a)?

FCFS P<sub>1</sub>=0 P<sub>2</sub>=10 P<sub>3</sub>=11 P<sub>4</sub>=13 P<sub>5</sub>=14

SJF P<sub>2</sub>=0 P<sub>4</sub>=1 P<sub>3</sub>=2 P<sub>5</sub>=4 P<sub>1</sub>=9

Prioritas Non-preemptive P<sub>2</sub>=0 P<sub>5</sub>=1 P<sub>1</sub>=6 P<sub>3</sub>=16 P<sub>4</sub>=18

RR P<sub>1</sub>=8 P<sub>2</sub>=1 P<sub>3</sub>=5 P<sub>4</sub>=3 P<sub>5</sub>=4

b. Pertimbangkan soal proses berikut, dg pengangkutan (PV) dan waktu keterbatasan dibentuk dlm matriks!

Process	Burst time	Priority
P <sub>1</sub>	0	0
P <sub>2</sub>	4	0,4
P <sub>3</sub>	1	1

a. Buat empat diagram Gantt yg menggambarkan eksekusi proses2 ini menggunakan algoritma pengjadualan berikut: FCFS, SJF, (algoritma dapat dihitung ke dlm ms menurut proses yg lbh prioritas akhir dlm).

FCFS | P<sub>1</sub> | P<sub>2</sub> | P<sub>3</sub> |

0 8 12 13

SJF | P<sub>3</sub> | P<sub>2</sub> | P<sub>1</sub> |

0 1 5 13

Prioritas Non-preemptive | P<sub>1</sub> | P<sub>2</sub> | P<sub>3</sub> |

0 8 12 13

RR (Round Robin, Quantum=1) | P<sub>1</sub> | P<sub>2</sub> | P<sub>3</sub> | P<sub>1</sub> | P<sub>2</sub> | P<sub>1</sub> | P<sub>2</sub> | P<sub>1</sub> | P<sub>2</sub> |

0 1 2 3 4 5 6 7 8 13

b. Berapa waktu tunggu untuk setiap proses utk masing2 algoritma pengjadualan pt bagian (a)?

$$FCFS = P_1 = 0 \quad P_2 = 12 \quad P_3 = 13$$

$$SJF = P_3 = 1 \quad P_2 = 5 \quad P_1 = 13$$

$$\text{Prioritas Non-preemptive} = P_1 = 0 \quad P_2 = 12 \quad P_3 = 13$$

$$RR = P_1 = 13 \quad P_2 = 6 \quad P_3 = 3$$

c. Berapa waktu tunggu utk setiap proses utk masing2 algoritma pengjadualan pt bagian (a)?

$$FCFS = P_1 = 0 \quad P_2 = 8 \quad P_3 = 12$$

$$SJF = P_3 = 0 \quad P_2 = 1 \quad P_1 = 5$$

$$\text{Prioritas Non-preemptive} = P_1 = 0 \quad P_2 = 8 \quad P_3 = 12$$

$$RR = P_1 = 5 \quad P_2 = 2 \quad P_3 = 2$$

Jelaskan perbedaan dlm tingkat yg diberikan oleh algoritma pengjadualan

bentuk berhadip proses yg pendek:

a. First Come First Served

Tdk memdiskriminasi secara langsung tgl proses yg pendek sehingga proses yg pendek bisa terbuka dlm proses yg panjang.

b. Round Robin :

Membentuk jmlh waktu yg sama pd semua proses, sehingga lbh responsif berhadip proses pendek dibandingkan FCFS, tetapi tdk memdiskriminasi secara eksplisit.

c. Multilevel Feedback Queues

Sangat memdiskriminasi tgl proses pendek, dg memprioritaskan proses dlm antren dg prioritas lbh tinggi.

d. Tgl secara singkat bentuk ini:

a. Waiting time

Waktu tunggu = total waktu yg dibutuhkan oleh suatu proses dr dlm antren sdh sebelum proses tsb mulai dijalankan oleh CPU.

b. Response time

Waktu respon = waktu yg dibutuhkan oleh kebutuhan proses dlm sistem hingga pertama kali tiberikan ke CPU.

c. Throughput

Throughput = jumlah total proses yg diselesaikan oleh sistem dlm periode waktu tertentu.