

# Pertemuan 3

## PENJADWALAN PROSES

# DESKRIPSI PENJADWALAN PROSES

Penjadwalan merupakan kumpulan kebijaksanaan dan mekanisme di sistem operasi yang berkaitan dengan urutan kerja yang dilakukan sistem komputer.

Penjadwalan bertugas memutuskan hal-hal berikut:

1. Proses harus berjalan,
2. Kapan dan berapa lama proses itu berjalan.

Sasaran utama penjadwalan proses adalah optimasi kinerja sistem komputer menurut kriteria tertentu.

# Kriteria Penjadwalan

- ❑ Adil (Fairness) adalah proses-proses diperlakukan sama yaitu mendapat jatah waktu layanan pemroses yang sama dan tidak ada proses yang tidak kebagian layanan pemroses.
- ❑ Efisiensi (Utilisasi) adalah pemroses yang dihitung dengan perbandingan (rasio) waktu sibuk pemroses dengan total waktu operasi sistem komputer.
- ❑ Waktu Tanggap (Response Time)
  - Sistem Interaktif didefinisikan sebagai waktu yang dihabiskan dari saat katakter terakhir dari perintah dimasukkan oleh program sampai hasil pertama muncul di perangkat masukan keluaran seperti layar.
  - Sistem waktu nyata didefinisikan sebagai waktu dari saat kemunculan suatu kejadian sampai instruksi pertama rutin layanan terhadap kejadian dieksekusi.

## Kriteria Penjadwalan (Lanjut...)

- ❑ *Turn Around Time* adalah waktu yang dihabiskan dari saat proses atau *job* mulai masuk ke sistem sampai proses itu diselesaikan sistem. Waktu yang dimaksud adalah waktu yang dihabiskan proses berada di sistem, diekspresikan sebagai penjumlahan waktu eksekusi dan waktu menunggu dari proses itu, yaitu:

***Turn around time* = waktu eksekusi + waktu menunggu**

- ❑ *Throughput* adalah jumlah kerja yang dapat diselesaikan selama satu selang/unit waktu. Cara untuk mengekspresikan *throughput* adalah dengan jumlah proses/*job* pemakai yang dapat dieksekusi dalam satu unit/interval waktu tertentu.

# Tipe-tipe penjadwalan :

## 1. Penjadwalan jangka pendek

Bertugas menjadwalkan alokasi pemroses di antara proses-proses ready di memori utama. Penjadwalan dijalankan setiap terjadi pengalihan proses untuk memilih proses berikutnya yang harus dijalankan.

## 2. Penjadwalan jangka menengah

Proses status ready – running – waiting – ready

## 3. Penjadwalan jangka panjang

Penjadwal ini bekerja terhadap antrian batch dan memilih batch berikutnya yang harus dieksekusi. Batch biasanya adalah proses-proses dengan penggunaan sumber daya yang intensif (yaitu waktu pemroses, memori, perangkat masukan/keluaran), program-program ini berprioritas rendah, digunakan sebagai pengisi (agar pemroses sibuk) selama periode aktivitas job-job interaktif rendah

# Penjadwalan proses

3 istilah yang digunakan pada penjadwalan proses

## 1. Antrian

Sejumlah proses yang menunggu menggunakan prosesor dan akan diproses sesuai dengan urutan antrian proses.

Proses berada dalam memori utama

# Penjadwalan proses (Lanjut....)

## 2. Prioritas

Mendahului pada antrian proses, kalau proses itu berada di bagian belakang antrian, maka dengan pemberian prioritas, proses itu langsung berada di bagian paling depan pada antrian itu sambil menunggu sampai kerja prosesor selesai.

## 3. Prempsi

Mendahului pada antrian proses, kalau proses itu berada di bagian belakang antrian, maka dengan pemberian prempsi, proses itu langsung berada di bagian paling depan pada antrian itu bahkan akan memberhentikan kerja prosessor untuk mengerjakan proses yang prempsi tersebut.

# Perhitungan pada Kerja Prosesor

## Lama proses ( $t$ )

adalah lama waktu yang diperlukan untuk mengolah proses itu di dalam prosesor

## Lama tanggap ( $T$ )

adalah Waktu yang diperlukan untuk proses sejak mulai sampai selesai di olah oleh prosesor

Terdapat 2 macam lama tanggap :

- ❑ *Turn around time*, Dengan memperhitungkan lama waktu yang digunakan untuk sebuah proses hingga keluaran.
- ❑ *Response time*, Tidak memperhitungkan lama waktu yang digunakan untuk sebuah proses hingga keluaran



# Rumus Penjadwalan

Jika terdapat  $N$  proses serentak, serta setiap proses memiliki lama tanggap sebesar  $T$ , maka rerata lama tanggap  $Tr$  adalah

$$Tr = (\text{jumlah } Ti) / N$$

- Waktu sia-sia ( $T - t$ ), waktu yang terbuang dalam antrian atau selama terkena Prempsi.
- Rasio tanggap ( $Rt$ ), Perbandingan di antara lama proses terhadap lama tanggap
- Rasio pinalti ( $Rp$ ), Perbandingan diantara lama tanggap terhadap lama proses.

$$Rt = \frac{t}{T}$$

dan

$$Rp = \frac{T}{t}$$

# Algoritma Penjadwalan

## Algoritma Penjadwalan Pertama tiba pertama dilayani (PTPD/ FCFS/FIFO)

Penjadwalan tanpa prioritas tanpa prempsi, Proses yang tiba lebih dahulu akan dilayani lebih dahulu, jika tiba pada waktu yang bersamaan akan dilayani sesuai dengan urutan pada antrian.

# Contoh Algoritma PTPD/FCFS/FIFO

Contoh Algoritma PTPD/FCFS/FIFO sebagai berikut:

Nama proses	Saat tiba	Lama proses
A	0	4
B	1	7
C	3	3
D	7	8

# Penyelesaian Algoritma PTPD/FCFS/FIFO

Nama Proses	Saat Tiba	Lama Proses	Saat Mulai	Saat Rampung	Lama tanggap
A	0	4	0	4	4
B	1	7	4	11	10
C	3	3	11	14	11
D	7	8	14	22	15
				Jumlah	40
				Rerata	10

Barisan proses :

Saat	0	1	2	3	4	5	6	7	8	9	10
Proses	A	A	A	A	B	B	B	B	B	B	

Saat	10	11	12	13	14	15	16	17	18	19	20
Proses	B	C	C	C	D	D	D	D	D	D	
Saat	20	21	22	23	24	25	26	27	28	29	30
Proses	D	D									

## **3.2. Algoritma Penjadwalan Proses Terpendek Dipertamakan (PTD/ SJF/ SJN)**

Penjadwalan dengan prioritas tanpa prempsi, Terdapat 2 langkah :

Penentuan berdasarkan pendeknya proses yang dilayani  
Jika proses yang terpendek tersebut belum tiba maka prosesor akan melayani proses yang telah tiba sampai proses tersebut selesai.

# Contoh Algoritma Penjadwalan Proses Terpendek Dipertamakan (PTD/ SJF/ SJN)

Contoh :

Tabel proses PTD

Nama proses	Saat tiba	Lama proses
A	0	5
B	3	7
C	5	2
D	6	4

# Penyelesaian Algoritma Penjadwalan Proses Terpendek Dipertamakan (PTD/ SJF/ SJN)

Tabel PTD

Nama Proses	Saat Tiba	Lama Proses	Saat Mulai	Saat Rampung	Lama tanggap
A	0	5	0	5	5
B	3	7	11	18	15
C	5	2	5	7	2
D	6	4	7	11	5
Jumlah					27
Rerata					6,75

Barisan proses :

Saat	0	1	2	3	4	5	6	7	8	9	10
Proses	A	A	A	A	A	C	C	D	D	D	

Saat	10	11	12	13	14	15	16	17	18	19	20
Proses	D	B	B	B	B	B	B	B			

# Dosen memberikan contoh latihan lainnya