

# Praktikum 10

## Penjadwalan CPU 2

### POKOK BAHASAN:

- ✓ Membuat program simulasi Pendawalan CPU

### TUJUAN BELAJAR:

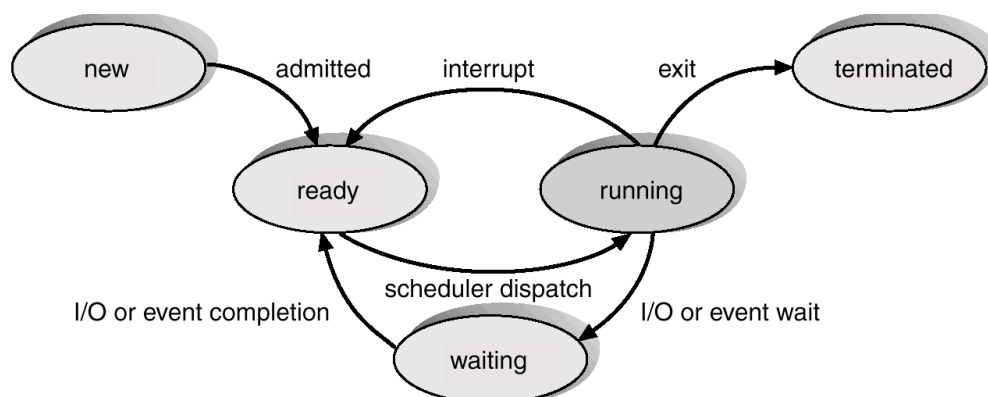
Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:

- ✓ Memahami cara Penjadwalan CPU.
- ✓ Membuat Aplikasi simulasi Algoritma penjadwalan first come first serve
- ✓ Membuat Aplikasi simulasi Algoritma penjadwalan shortest job first
- ✓ Membuat Aplikasi simulasi Algoritma penjadwalan round robin

### DASAR TEORI:

#### 1 Penjadwalan CPU Preemptive

Penjadual CPU preventive adalah penjadwalan cpu yang dapat disela walupun proses masih dalam proses pengerjaan. Perhatikan diagram proses pada gambar 10-1 :



Gambar 10 -1 . Diagram Proses.

Kapan pun CPU menjadi idle, sistem operasi harus memilih salah satu proses untuk masuk kedalam antrian ready (siap) untuk dieksekusi. Pemilihan tersebut dilakukan oleh penjadual short term. Penjadual memilih dari sekian proses yang ada di memori yang sudah siap dieksekusi, dan mengalokasikan CPU untuk mengeksekusinya. Penjadual CPU mungkin akan dijalankan ketika proses:

1. Berubah dari running ke waiting state.
2. Berubah dari running ke ready state.
3. Berubah dari waiting ke ready.
4. Terminates.

Penjadual dari no 1 sampai 4 *non preemptive sedangkan* yang lain *preemptive*. Dalam penjadual nonpreemptive sekali CPU telah dialokasikan untuk sebuah proses, maka tidak bisa di ganggu, penjadual model seperti ini digunakan oleh Windows 3.x; Windows 95 telah menggunakan penjadual preemptive.

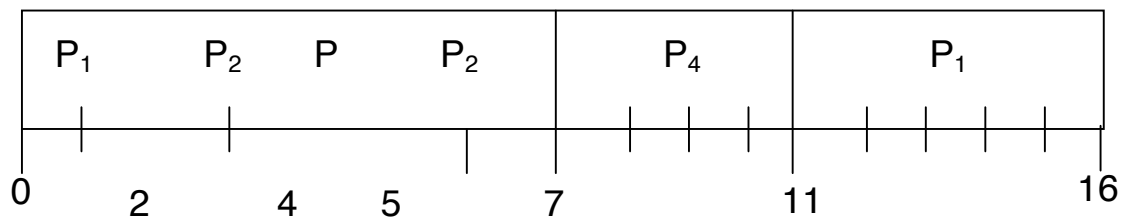
## 2 Penjadwalan Shortest Job First Preemptive

SJF preventive bila sebuah proses datang dengan waktu proses lebih rendah dibandingkan dengan waktu proses yang sedang dieksekusi oleh CPU maka proses yang waktunya lebih rendah mendapatkan prioritas. Skema ini disebut juga Short - Remaining Time First (SRTF).

Contoh:

<u>Arrival Time</u>	<u>Burst Time</u>	<u>proses</u>
0.0	7	P1
2.0	4	P2
4.0	1	P3
5.0	4	P4

**Gambar 10-4. Kedatangan Proses.**



$$\text{Rata-rata waktu tunggu} = (9 + 1 + 0 + 2)/4 = 3$$

### 3 Penjadual Prioritas

Penjadualan SJF (Shortest Job First) adalah kasus khusus untuk algoritma penjadual Prioritas. Prioritas dapat diasosiasikan masing-masing proses dan CPU dialokasikan untuk proses dengan prioritas tertinggi. Untuk prioritas yang sama dilakukan dengan FCFS.

Ada pun algoritma penjadual prioritas adalah sebagai berikut:

- Setiap proses akan mempunyai prioritas (bilangan integer). Beberapa sistem menggunakan integer dengan urutan kecil untuk proses dengan prioritas rendah, dan sistem lain juga bisa menggunakan integer urutan kecil untuk proses dengan prioritas tinggi. Tetapi dalam teks ini diasumsikan bahwa integer kecil merupakan prioritas tertinggi.
- CPU diberikan ke proses dengan prioritas tertinggi (integer kecil adalah prioritas tertinggi).
- SJF adalah contoh penjadual prioritas dimana prioritas ditentukan oleh waktu pemakaian CPU

Beikutnya. Permasalahan yang muncul dalam penjadualan prioritas adalah indefinite blocking atau starvation.

- Kadang-kadang untuk kasus dengan prioritas rendah mungkin tidak pernah dieksekusi. Solusi untuk algoritma penjadual prioritas adalah aging
- Prioritas akan naik jika proses makin lama menunggu waktu jatah CPU.

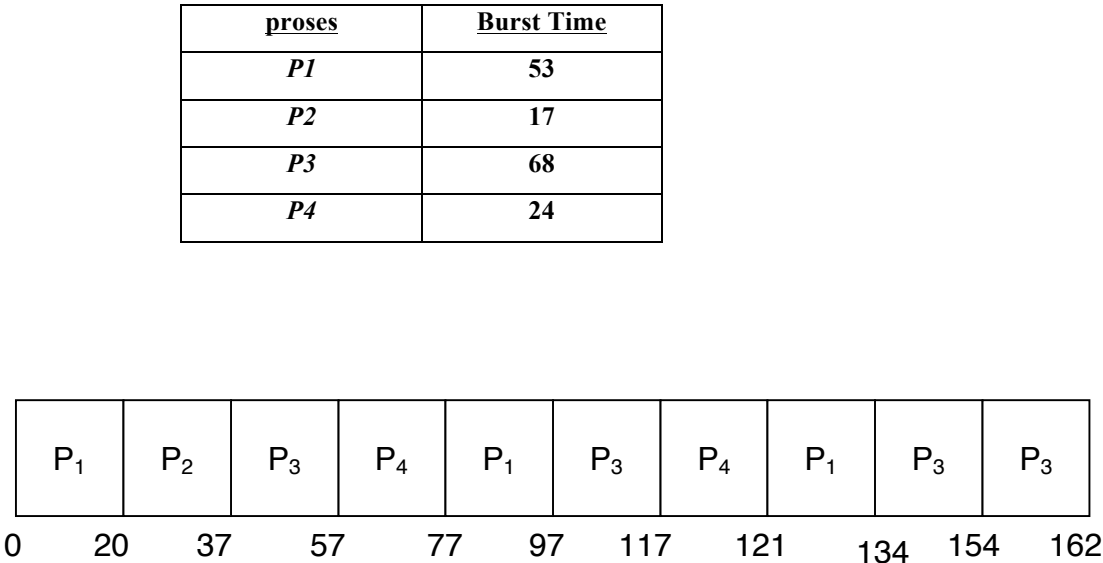
## 4 Penjadual Round Robin

Algoritma Round Robin (RR) dirancang untuk sistem time sharing. Algoritma ini mirip dengan penjadual FCFS, namun preemption ditambahkan untuk switch antara proses. Antrian ready diperlakukan atau dianggap sebagai antrian sirkular. CPU mengililingi antrian ready dan mengalokasikan masing-masing proses untuk interval waktu tertentu sampai satu time slice/ quantum.

Berikut algoritma untuk penjadual Round Robin:

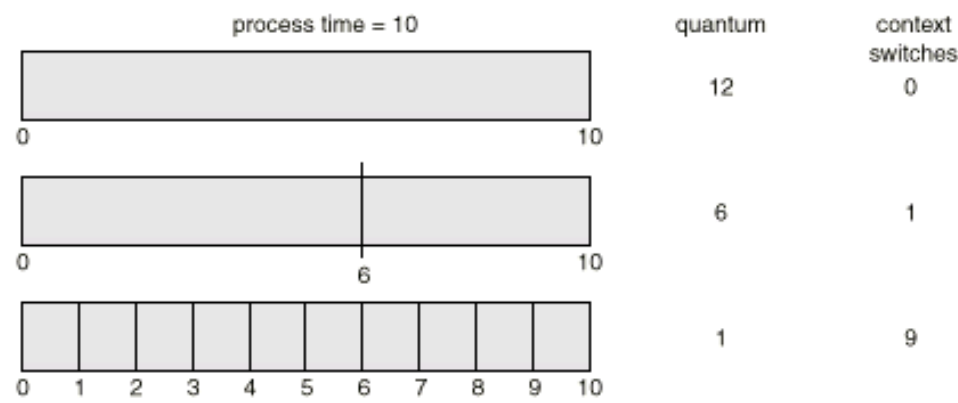
- Setiap proses mendapat jatah waktu CPU (time slice/ quantum) tertentu. Time slice/quantum umumnya antara 10 - 100 milidetik.
  1. Setelah time slice/ quantum maka proses akan di-preempt dan dipindahkan ke antrian ready.
  2. Proses ini adil dan sangat sederhana.
- Jika terdapat  $n$  proses di "antrian ready" dan waktu quantum  $q$  (milidetik), maka:
  1. Maka setiap proses akan mendapatkan  $1/n$  dari waktu CPU.
  2. Proses tidak akan menunggu lebih lama dari:  $(n-1)q$  time units.
- Kinerja dari algoritma ini tergantung dari ukuran time quantum
  1. Time Quantum dengan ukuran yang besar maka akan sama dengan FCFS
  2. Time Quantum dengan ukuran yang kecil maka time quantum harus diubah ukurannya lebih besar dengan respek pada alih konteks sebaliknya akan memerlukan ongkos yang besar.

Tipikal: lebih lama waktu rata-rata turnaround dibandingkan SJF, tapi mempunyai response terhadap user lebih cepat.



Gambar 2-38. Round Robin

Time Quantum Vs Alih Konteks



Gambar 2-39. Time Quantum dan Alih Konteks.

TUGAS PENDAHULUAN:

1. Lihat Tabel dibawah,, Gambarkan 4 diagram Chart yang mengilustrasikan eksekusi dari proses-proses tersebut menggunakan FCFS, SJF nonpreemptive, round robin.

Tabel Proses

Proses	Burst Time	Prioritas
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

Keterangan:

Misal diberikan beberapa proses dibawah ini dengan panjang CPU burst ( dalam milidetik)

Semua proses diasumsikan datang pada saat  $t=0$  dan quantum time= 3

### **PERCOBAAN:**

1. Bukalah program (boleh menggunakan bahasa apa saja pemrograman yang sudah di pelajari)

#### **Percobaan 1 : Priority**

1. Buatlah program untuk menampilkan diagram chart untuk penjadwalan priority

#### **Percobaan 2 : Round Robin**

1. Buatlah program untuk menampilkan diagram chart untuk penjadwalan Round Robin

#### **Percobaan 3 Shortest Job First Preemptive**

1. Buatlah program untuk menampilkan diagram chart untuk penjadwalan Shortest Job First Preemptive

**LAPORAN RESMI:**

1. Analisa hasil percobaan yang Anda lakukan.
2. Buatlah Program di atas.
3. Berikan kesimpulan dari praktikum ini.