



PENJADWALAN SISTEM OPERASI

Definisi

- **Penjadwalan proses** merupakan kumpulan kebijaksanaan dan mekanisme di sistem operasi yang berkaitan dengan urutan kerja yang dilakukan sistem komputer.
- Sedangkan **proses** sendiri merupakan unit kerja terkecil yang secara individu memiliki sumberdaya atau unit pemilikan sumberdaya.

Tugas Penjadwalan :

- Memutuskan proses yang harus berjalan
- Memutuskan kapan dan selama berapa lama proses itu berjalan

Sasaran

- ✓ *Efisiensi*

Efisiensi atau utilisasi pemroses dihitung dengan perbandingan (rasio) waktu sibuk pemroses.

- ✓ *Turn Around Time*

Adalah waktu yang dihabiskan dari saat program atau job mulai masuk ke sistem sampai proses diselesaikan sistem. Waktu yang dimaksud adalah waktu yang dihabiskan di dalam sistem. Sasaran penjadwalan adalah meminimalkan turn around time.

Turn around time = waktu eksekusi + waktu menunggu

- ✓ *Throughput*

Adalah jumlah kerja atau jumlah job yang dapat diselesaikan dalam satu unit waktu. Sasaran penjadwalan adalah memaksimalkan jumlah job yang diproses per satu interval waktu. Lebih tinggi angka throughput, lebih banyak kerja yang dilakukan sistem.

Berdasarkan Tugas

- **Penjadwalan jangka pendek (short-termscheduller)**

Penjadwalan ini bertugas menjadwalkan alokasi pemroses diantara proses-proses ready di memori utama.

- **Penjadwalan jangka menengah (medium termscheduller)**

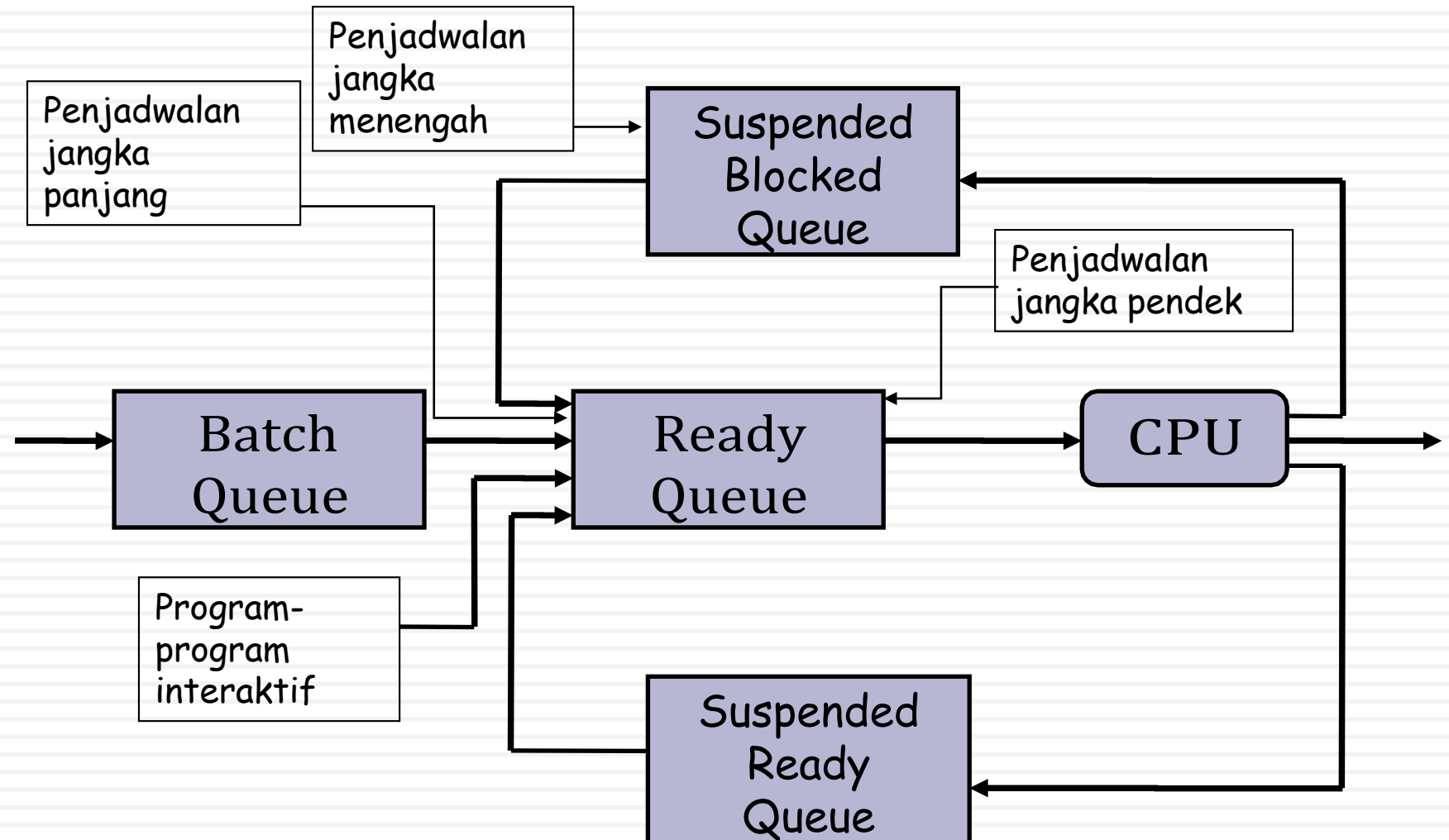
Penjadwalan jangka menengah adalah menangani proses-proses swapping (aktivitas pemindahan proses yang tertunda dari memory utama ke memory sekunder).

- **Penjadwalan jangka panjang (long-termscheduller)**

Penjadwalan jangka panjang bekerja terhadap antrian batch (proses – proses dengan penggunaan sumberdaya yang intensif) dan memilih batchberikutnya yang harus di eksekusi.

Berdasarkan Tugas

Menujukan posisi dari tipe-tipe penjadwalan yang terdapat pada satu sistem operasi



Jenis Strategi Penjadwalan

□ Penjadwalan *Nonpreemptive*

Begitu proses diberi jatah waktu pemroses maka pemroses tidak dapat diambil alih oleh proses lain sampai proses itu selesai

□ Penjadwalan *Preemptive*

Saat proses diberi jatah waktu pemroses maka pemroses dapat diambil alih oleh proses lain sehingga proses disela sebelum selesai dan harus dilanjutkan menunggu jatah waktu pemroses tiba kembali pada proses itu

Algoritma Penjadwalan bdsk Strategi

Algoritma – algoritma yang menerapkan strategi *nonpreemptive* :

- ❑ **FIFO** (*First-in, First-out*) atau **FCFS** (First Come, First Serve).
- ❑ **SJF** (*Shortest Job First*).
- ❑ **HRN** (*Highest Ratio Next*).
- ❑ **MFQ** (*Multiple Feedback Queues*).

Algoritma – algoritma yang menerapkan strategi *preemptive* :

- ❑ **RR** (*Round-Robin*).
- ❑ **SRF** (*Shortest-Remaining-First*).
- ❑ **PS** (*Priority Scheduling*).
- ❑ **GS** (*Guaranteed Scheduling*).

Algoritma Penjadwalan bdsk Prioritas

Klasifikasi lain berdasarkan adanya prioritas di proses - proses, yaitu :

1. **Algoritma penjadwalan tanpa berprioritas.**
2. **Algoritma penjadwalan berprioritas, terdiri dari :**
 - Algoritma penjadwalan berprioritas statik.
 - Algoritma penjadwalan berprioritas dinamis.

Non Preemptive

Penjadwalan FIFO (First In First Out)

Merupakan penjadwalan nonpreemptive dan penjadwalan tidak berprioritas. Penjadwalan FIFO adalah penjadwalan paling sederhana, yaitu :

- Proses-proses diberi jatah waktu pemroses berdasarkan waktu kedatangan
- Saat proses mendapat jatah waktu pemroses, proses dijalankan sampai selesai

Penjadwalan ini adil yaitu proses yang datang duluan, dilayani duluan juga. Dikatakan tidak adil karena job-job yang perlu waktu lama membuat job-job pendek menunggu. Job-job tak penting dapat membuat job-job penting menunggu.

Contoh Soal:

Jika diketahui 5 macam antrian proses yaitu : A-B-C-D-E dengan waktu kedatangan semuanya 0 dan prosesor sedang tidak aktif ($PP=0$), lama proses berturut-turut antara lain 5,2,6,8,3.

Penjadwalan FIFO

Nama Proses	Waktu Tiba	Lama Eksekusi
A	0	5
B	0	2
C	0	6
D	0	8
E	0	3

$$\Sigma TA = 70$$

$$\text{Rerata TA} = 14$$

Nama Proses (1)	Waktu Tiba (2)	Lama Eksekusi (3)	Mulai Eksekusi (4)	Selesai Eksekusi (5)	TA (6)
A	0	5	0	5	5
B	0	2	5	7	7
C	0	6	7	13	13
D	0	8	13	21	21
E	0	3	21	24	24

Penjadwalan FIFO (First In First Out)

- Terlihat lama tanggap : 14 Nilai ini cukup besar bila dibanding lama proses masing2.
- Contoh diatas tidak disertai waktu tiba.
- Berikut ini adalah contoh dengan waktu tiba yang berbeda:

Nama Proses	Waktu Tiba	Lama Eksekusi
A	0	5
B	1	2
C	2	6
D	2	8
E	5	3

Waktu Tunggu(TA) = selesai Eksekusi - Waktu Tiba

Penjadwalan FIFO (First In First Out)

Nama Proses 1	Waktu Tiba 2	Lama Eksekusi 3	Mulai Eksekusi 4	Selesai Eksekusi 5	TA 6
A	0	5	0	5	5
B	1	2	5	7	6
C	2	6	7	13	11
D	2	8	13	21	19
E	5	3	21	24	19
					$\sum TA = 60$ Rata2 = 12

Penjadwalan FIFO (First In First Out)

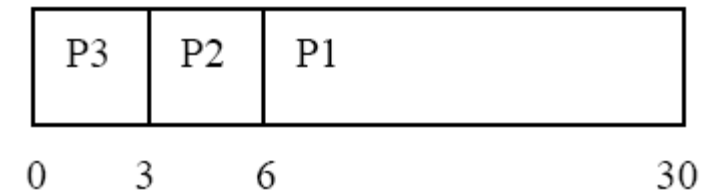
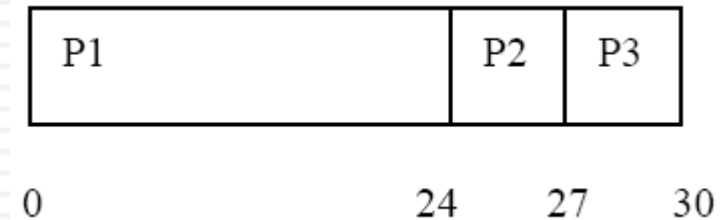
□ Cara lain untuk mengerjakan

- Misal ada tiga proses P1, P2, P3 yang datang dengan lama waktu kerja CPU (CPU Burst-time) masing-masing sbb :

Proses	Burst-time
P1	24
P2	3
P3	3

Dari Gantt Chart dapat diambil kesimpulan waktu tunggu untuk P1 adalah 0 milidetik, waktu tunggu untuk P2 adalah 24 milidetik, waktu tunggu P3 adalah 27 milidetik. **Jadi rata-rata waktu tunggu (Average Waiting Time / AWT) adalah $(0+24+27)/3 = 17$ milidetik.**

Kemudian jika waktu kedatangan proses adalah P3, P2, P1 maka Gantt Chartnya adalah



Penjadwalan FIFO (First In First Out)

- Menentukan Turn Around Time dengan FIFO berdasarkan contoh diatas :



- Turn around time (waktu penyelesaian) P1 adalah 24, P2 = 27, P3 = 30, maka rata-rata turn around time = $(24+27+30)/3 = 27$ milidetik.

Penjadwalan Shortest Job First (SJF)

Penjadwalan dengan tipe prioritas tanpa preeemptive.

- Dasar prioritasnya adalah pendeknya proses.
- Makin pendek prosesnya makin tinggi prioritasnya.

Nama Proses	Waktu Tiba	Lama Eksekusi
A	0	10
B	0	5
C	0	7
D	0	1
E	0	3

Nama Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	TA
A	0	1	0	1	1
B	0	3	1	4	4
C	0	5	4	9	9
D	0	7	9	16	16
E	0	10	16	26	26
SISTEM OPERASI – Metode Penjadwalan Proses				$\Sigma TA = 56$ Rerata = 11.2	

Waktu tiba =
sama (0)

Penjadwalan Shortest Job First (SJF)

SJF dengan waktu tiba yang berbeda

Nama Proses (1)	Waktu Tiba (2)	Lama Eksekusi (3)	Mulai Eksekusi (4)	Selesai Eksekusi (5)	TA (6)
A	0	1	0	1	1
C	5	3	5	8	3
B	2	5	7	12	10
E	9	7	16	23	14
D	7	10	23	33	26

SISTEM OPERASI – Metode Penjadwalan Proses

$\Sigma TA = 54$
Rerata = 10.8

Nama Proses	Waktu Tiba	Lama Eksekusi
A	0	1
B	2	5
C	5	3
D	7	10
E	9	7

Penjadwalan Highest Ratio Next (HRN)

Penjadwalan ini merupakan :

- Penjadwalan *non-preemptive*
- Penjadwalan berprioritas dinamis.

Rumus : $R_p = (s + t) / t$
Lama tunggu : s, lama proses : t

Nama Proses	Waktu Tiba	Lama Eksekusi
A	0	4
B	1	2
C	2	5
D	3	8
E	4	4

Dari soal diatas yang dikerjakan pertama kali adalah job yang tiba pertama kali. Dikarenakan job lain belum tiba.

Nama Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	TA
A	0	4	0	4	4

Penjadwalan Highest Ratio Next (HRN)

Jawab tabel rasio penalti ke 1

PROSES	WAKTU SIA-SIA	RASIO PENALTI
B	3	$(3+2)/2=2,5$
C	2	$(2+5)/5=1,2$
D	1	$(1+8)/8=1,125$
E	0	$(0+4)/4=1$

Rasio penalti ke 1 tertinggi adalah job B, Maka tabel proses ke 2 adalah

Nama Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	TA
A	0	4	0	4	4
B	1	2	4	6	5

Pada tabel 2 didapat proses B selesai pada saat 6 maka tabel rasio penalti ke 2 adalah

Penjadwalan Highest Ratio Next (HRN)

PROSES	WAKTU SIA-SIA	RASIO PENALTI
C	4	$(4+5)/5=1,8$
D	3	$(3+8)/8=1,375$
E	2	$(2+4)/4=1,5$

- Pada saat 6 proses C punya rasio penalti tertinggi, jadi proses C dikerjakan sampai selesai pada saat 11 yaitu

Nama Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	TA
A	0	4	0	4	4
B	1	2	4	6	5
C	2	5	6	11	9

Penjadwalan Highest Ratio Next (HRN)

- Pada rasio penalti proses menjadi (tabel rasio penalti ke 3)

PROSES	WAKTU SIA-SIA	RASIO PENALTI
D	8	$(8+8)/8=2$
E	7	$(7+4)/4=2,75$

- Pada saat 11 ini, proses E punya rasio tertinggi sehingga proses E yang dikerjakan, pada saat 15 proses E selesai, dan mulai dari 15 proses D dikerjakan sampai selesai saat 23

Nama Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	TA
A	0	4	0	4	4
B	1	2	4	6	5
C	2	5	6	11	9
E	4	4	11	15	11
D	3	8	15	23	20
				$\Sigma TA = 49$ Rerata = 9,8	

Penjadwalan Multiple Feedback Queues (MFQ)

Penjadwalan ini merupakan :

- Penjadwalan *preemptive (by time)* dan berprioritas dinamis.

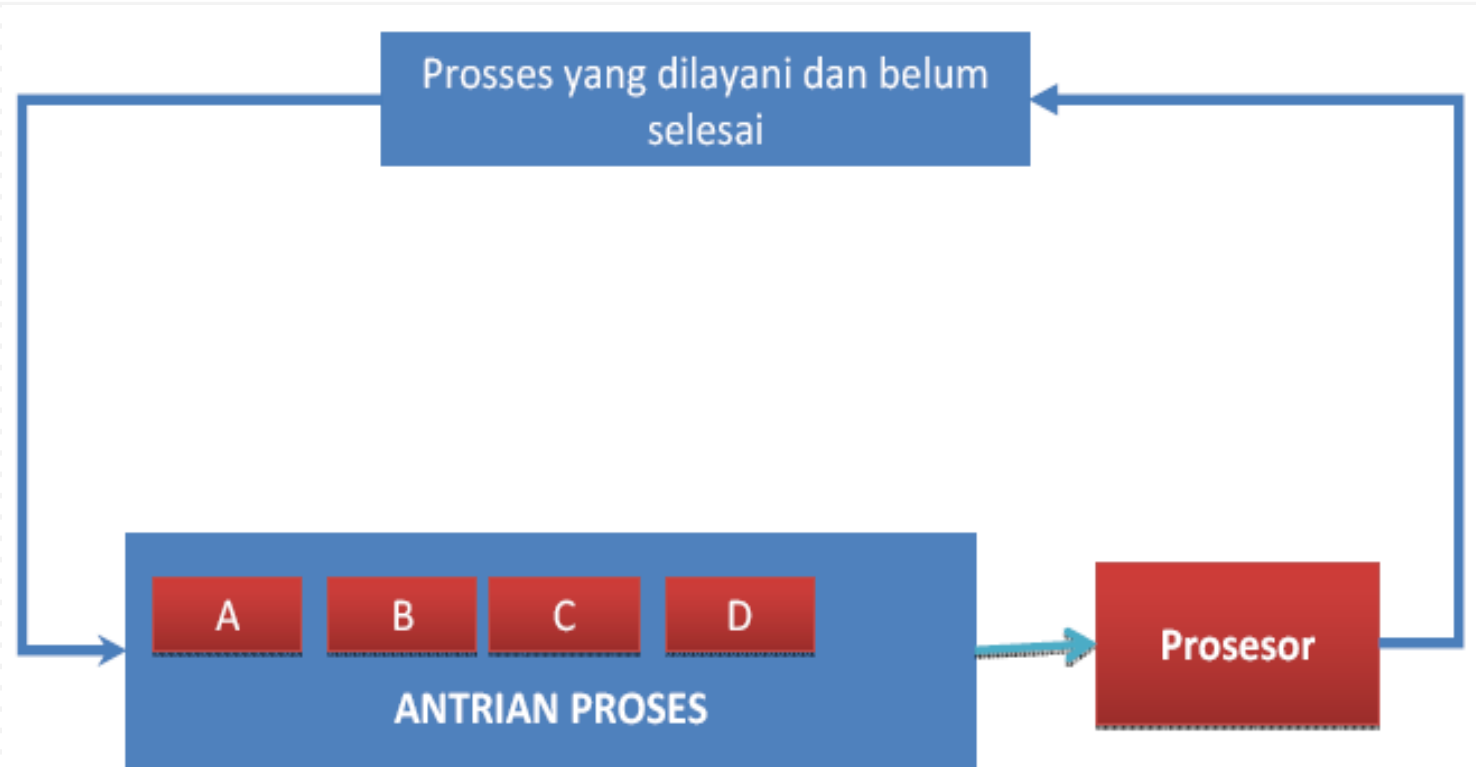
Latihan 1. Penjadwalan Non Preemptive

1. Diketahui 10 antrian proses yaitu : A-B-C-D-E-F-G-H-I-J dengan waktu kedatangan semuanya 0 dan prosesor sedang tidak aktif ($PP=0$), lama proses berturut-turut antara lain 8,5,4,10,3,9,2,1,7,8 (Menggunakan FIFO).
2. Diketahui 7 antrian proses yaitu : P-Q-R-S-T-U-V dengan waktu kedatangan semuanya 0 dan prosesor sedang tidak aktif ($PP=0$), lama proses berturut-turut antara lain 6,9,1,5,7,7,2 (Menggunakan SJF).
3. Diketahui 5 antrian proses yaitu : A-B-C-D-E dengan waktu kedatangan semuanya 0, 1, 2, 3,4 dan prosesor sedang tidak aktif ($PP=0$), lama proses berturut-turut antara lain 1,5,7,9,10 (HRN)

Penjadwalan Preemptive

□ Penjadwalan Round Robin (RR)

- Pada penjadwalan RR ini Eksekusi dijalankan secara giliran berdasarkan antrian (Non - Preemptive), prosesor mengerjakan sesaat setiap proses secara berturut-turut.
- Proses yang telah dieksekusi tapi belum selesai akan kembali ke antrian terakhir.



Penjadwalan Round Robin

NAMA PROSES	WAKTU TIBA	LAMA EKSEKUSI
A	0	7
B	0	5
C	0	8
D	0	2
E	0	6
Quantum (Q)=3		

NAMA PROSE	WAKTU TIBA	LAMA EKSEKUSI	MULAI EKSEKUSI	SELESAI EKSEKUSI	TA
A	0	7	0	26	26
B	0	5	3	19	19
C	0	8	6	28	28
D	0	2	9	11	11
E	0	6	11	25	25
SISTEM OPERASI – Metode Penjadwalan Proses				$\Sigma TA = 109$ Rerata = 21,8	

Penjadwalan Round Robin (RR)

Jika waktu tiba proses bukan =0, maka:

NAMA PROSES	WAKTU TIBA	LAMA EKSEKUSI
A	0	5
B	1	3
C	5	7
D	6	1
E	7	6
Quantum (Q)=2		

NAMA PROSE	WAKTU TIBA	LAMA EKSEKUSI	MULAI EKSEKUSI	SELESAI EKSEKUSI	TA
A	0	5	0	11	11
B	1	3	2	7	6
C	5	7	7	22	17
D	6	1	9	10	4
E	7	6	11	21	14
SISTEM OPERASI – Metode Penjadwalan Proses				$\Sigma TA = 52$ Rerata = 10,4	

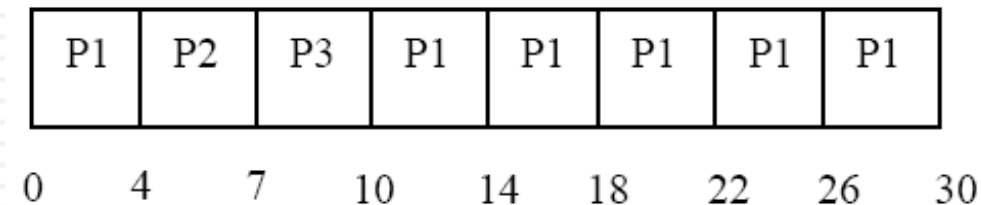


Penjadwalan Round Robin (RR)

- Contoh :
- Misal kumpulan proses datang pada waktu 0 dengan spesifikasi :

Proses	Burst-time
P1	24
P2	3
P3	3

Jika digunakan quantum time 4 milidetik, maka proses P1 mendapat 4 milidetik yang pertama, 20 milidetik berikutnya akan disela oleh proses P2 dan P3 secara bergantian, sehingga Gantt Chart-nya dapat digambarkan sbb:



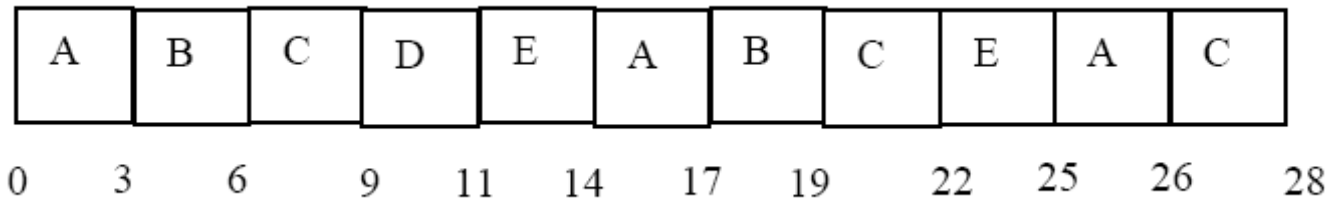
Waktu tunggu untuk tiap-tiap proses :

AWT yang terjadi adalah $(6+4+7)/3 = 5,66$ milidetik

Proses	Waiting Time
P1	$0+(10-4)=6$
P2	4
P3	7

Penjadwalan Round Robin (RR)

- Contoh :
- Menentukan Turn Around Time untuk quantum waktu (q) = 3



Proses	Saat Tiba	Lama Proses
A	0	7
B	0	5
C	0	8
D	0	2
E	0	6

Proses	Saat Tiba	Lama Proses	Saat Mulai	Saat Selesai	Turn Around Time
A	0	7	0	26	26
B	0	5	3	19	19
C	0	8	6	28	28
D	0	2	9	11	11
E	0	6	11	25	25
				Jumlah Rata-rata	109 21,8

Latihan 2

□ **Kerjakan 2 buah soal di bawah ini !**

1. Diketahui 6 antrian proses yaitu : E-F-G-H-I-J dengan waktu kedatangan berturut-turut 0,2,3,5,7,9 dan prosesor sedang tidak aktif ($PP=0$), lama proses berturut-turut antara lain 1,5,7,9,10,12 (**Menggunakan SJF dan HRN**).cari TAT dan AWT
2. Ada Lima proses (P1, P2, P3, P4 dan P5) tiba dalam saat yang berurutan pada 1, 3, 7, 8, 12. Hitung Turn Around Time (TAT) dan Average Waiting Time (AWT) dari kelima proses tersebut jika menggunakan Algoritma Penjadwalan Round Robin, dengan quantum 2. Burst time tiap proses secara berurutan adalah 8, 4, 6, 10, 12

Penjadwalan Shortest-Remaining-First (SRF)

Penjadwalan ini merupakan :

- Penjadwalan *preemptive* dengan sisa waktu terpendek.
- Penjadwalan berprioritas dinamis.

Contoh soal:

NAMA PROSES	WAKTU TIBA	LAMA EKSEKUSI
A	0	7
B	2	3
C	4	9
D	5	4

NAMA PROSE	WAKTU TIBA	LAMA EKSEKUSI	MULAI EKSEKUSI	SELESAI EKSEKUSI	TA
A	0	7	0	14	14
B	2	3	2	5	3
C	4	9	15	23	19
D	5	4	5	9	4
				$\Sigma TA = 40$ Rerata = 10	

Penjadwalan Priority Scheduling (PS)

Tiap proses dilengkapi dengan prioritas. CPU dialokasikan untuk proses yang memiliki prioritas paling tinggi. Jika beberapa proses memiliki prioritas yang sama, maka akan digunakan algoritma FIFO.

Prioritas dapat diberikan secara :

- ❑ Prioritas statis (*static priorities*).
- ❑ Prioritas dinamis (*dynamic priorities*).

Prioritas Statis

Prioritas statis berarti prioritas tak berubah.

Prioritas Dinamis

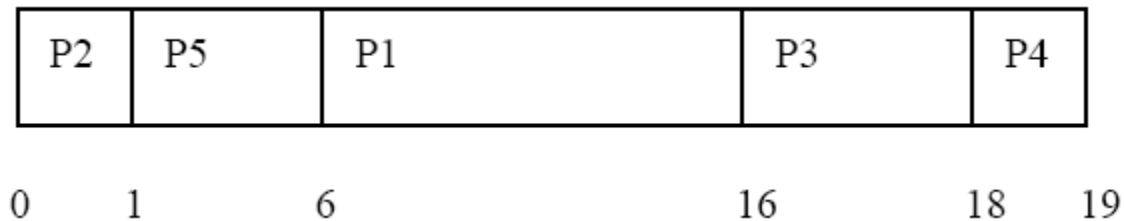
Merupakan mekanisme menanggapi perubahan lingkungan sistem beroperasi. Prioritas awal yang diberikan ke proses mungkin hanya berumur pendek setelah disesuaikan ke nilai yang lebih tepat sesuai lingkungan.

Penjadwalan Priority Scheduling (PS)

- Contoh :
- Jika ada 5 proses P1, P2, P3, P4 dan P5 dengan CPU burst :

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

Gant Chart:



Sehingga AWT = $(6+0+16+18+1) = 8,2$ ms.

Prioritas biasanya menyangkut masalah : waktu, memori yang dibutuhkan, banyaknya file yang boleh dibuka, dan perbandingan antara rata-rata I/O burst dengan rata-rata CPU burst.

Proses	Waiting Time
P1	6
P2	0
P3	16
P4	18
P5	1

Penjadwalan Guaranteed Scheduling (GS)

Penjadwalan ini harus menjamin bahwa algoritma tersebut mempunyai kinerja yang cukup bagus dan menjanjikan kelangsungan hidup yang baik. Contoh : misal ada n user yang sedang login, maka tiap-tiap user dijanjikan akan menerima $1/n$ dari kemampuan CPU.

Untuk meyakinkan bahwa setiap user mendapatkan jatah waktu menggunakan CPU sesuai dengan haknya maka sistem harus tahu berapa CPU time yang diperlukan oleh setiap proses dalam 1 user. Dan juga CPU time yang diperlukan oleh tiap-tiap user.

User	CPU Time
A	5
B	4
C	8
D	1
E	2

Penjadwalan Guaranteed Scheduling

- Total waktu yang dibutuhkan untuk mengakses kelima user tersebut adalah 20 ms, sehingga diharapkan tiap user mendapatkan $20/5 = 4$ ms. Pada kenyataannya, mulai dari login hingga saat ini tiap-tiap user telah mendapatkan CPU seperti terlihat pada tabel berikut. Dan rasio antara CPU yang diperoleh sampai saat ini (aktual) dengan CPU yang seharusnya diperoleh (4ms) dapat dicari :

Dapat dilihat bahwa user A memiliki rasio 0.75, artinya A baru mendapatkan $\frac{3}{4}$ dari jatah waktu yang seharusnya diterima. User B memiliki rasio 1.5, artinya B telah mendapatkan 1.5 waktu dari yang seharusnya ia dapatkan. Algoritma ini kemudian akan menjalankan proses dengan rasio yang paling rendah dahulu hingga proses tersebut mendapatkan rasio melebihi rasio proses yang sebelumnya punya rasio satu tingkat lebih tinggi darinya.

User	CPU aktual	Rasio
A	3	$\frac{3}{4} = 0.75$
B	6	$\frac{6}{4} = 1.5$
C	2	$\frac{2}{4} = 0.5$
D	1	$\frac{1}{4} = 0.25$
E	1	$\frac{1}{4} = 0.25$

Latihan 3

1. Terdapat 5 job yang datang hampir pada saat yang bersamaan. Estimasi waktu eksekusi (*burst time*) masing-masing 10, 6, 2, 4 dan 8 menit dengan prioritas masing-masing 3, 5, 2, 1 dan 4, dimana 5 merupakan prioritas tertinggi. Tentukan rata-rata *waktu turnaround* untuk penjadwalan CPU dengan menggunakan algoritma PS

2. Diketahui proses berikut :

Proses	Waktu eksekusi
P1	11
P2	4
P3	6
P4	16
P5	9

Diketahui waktu tiba proses diatas =0, tentukan TAT dan (AWT) menggunakan algoritma penjadwalan RR dengan quantum time=5,