

CPU Planlama (CPU Scheduling)

1. Geliş Zamanı (Arrival Time) – Sürecin hazır (ready) kuyruğuna giriş zamanı
2. Bekleme Süresi (Waiting Time) – Sürecin hazır kuyruğunda geçirdiği toplam süre
3. Cevap Süresi (Response Time) – Hazır kuyruğundaki sürecin CPU'yu ilk ele geçirdiği zaman
(cevap süresi = sürecin CPU'ya alındığı zaman – geliş zamanı)
4. Çalışma Süresi (Burst Time, Execution Time, Running Time) – Sürecin CPU'da geçirdiği toplam zaman
5. Tamamlanma Süresi (Completion Time, Exit Time) – Sürecin CPU'da çalışmasının bittiği zaman
6. Devir süresi (Turnaround Time) – Sürecin sistemde geçirdiği tüm zaman
(devir süresi = çalışma süresi + bekleme süresi)
(devir süresi = tamamlanma süresi – geliş zamanı)

Tüm bu süreler hesaplanırken I/O işlemi yapılmadığı varsayılır.

Önceliğe Göre Planlama (Priority Scheduling)

Avantajlar:

- Önce önemli işler çalıştırılır
- Gerçek zamanlı işletim sistemleri için kesintili modu en uygun sonuçları verir

Dezavantajlar:

- Daha düşük öncelikli işlere hiç sıra gelmeyebilir (kıtlık, starvation)
- Gerçekte işlerin ne kadar bekleyeceği, ne zaman CPU'ya alınacağı bilinmez

Örn: Önceliğe göre planlama (kesintili) (büyük sayı yüksek önceliği belirtmektedir)

PID	Geliş Zamanı	Çalışma Süresi	Öncelik
P1	0	4	2
P2	1	3	3
P3	2	1	4
P4	3	5	5
P5	4	2	5

P1	P2	P3	P4		P5	P2	P1
0	1	2	3	8	10	12	15

PID	Bitiş zamanı	Devir süresi	Bekleme süresi
P1	15	$15 - 0 = 15$	$15 - 4 = 11$
P2	12	$12 - 1 = 11$	$11 - 3 = 8$
P3	3	$3 - 2 = 1$	$1 - 1 = 0$
P4	8	$8 - 3 = 5$	$5 - 5 = 0$
P5	10	$10 - 4 = 6$	$6 - 2 = 4$

Ortalama devir süresi = $38 / 5 = 7,6$

Ortalama bekleme süresi = $23 / 5 = 4,6$

Round Robin

Doğası gereği kesintilidir. CPU belirli bir süre bir sürece atanır. Bu süreye **time quantum** veya **time slice** denir.

Avantajlar:

- En iyi ortalama cevap süresini verir
- Zaman paylaşımlı sistemler, istemci sunucu mimarisi, etkileşimli sistemler için idealdir

Dezavantajlar:

- Çalışma zamanı çok uzun süreçlerde kıtlık oluşabilir
- Performansı time quantuma doğrudan bağlıdır
- Süreçlere öncelik verilemez

time quantum küçüldükçe,

- Context switch sayısı artar
- Cevap süresi düşer
- Kıtlık olasılığı artar

Çok büyük **time quantum** ile FCFS'ye yaklaşır

Örn: Round Robin q=2

PID	Geliş Zamanı	Çalışma Süresi
P1	0	5
P2	1	3
P3	2	1
P4	3	2
P5	4	3

P1	P2	P3	P1	P4	P5	P2	P1	P5	
0	2	4	5	7	9	11	12	13	14

PID	Bitiş zamanı	Devir süresi	Bekleme süresi
P1	13	$13 - 0 = 13$	$13 - 5 = 8$
P2	12	$12 - 1 = 11$	$11 - 3 = 8$
P3	5	$5 - 2 = 3$	$3 - 1 = 2$
P4	9	$9 - 3 = 6$	$6 - 2 = 4$
P5	14	$14 - 4 = 10$	$10 - 3 = 7$

Ortalama devir süresi = $43 / 5 = 8,6$

Ortalama bekleme süresi = $29 / 5 = 5,8$

Örn: Round Robin q=3

PID	Geliş Zamanı	Çalışma Süresi
P1	5	5
P2	4	6
P3	3	7
P4	1	9
P5	2	2
P6	6	3

	P4	P5	P3	P2	P4	P1	P6	P3	P2	P4	P1	P3	
0	1	4	6	9	12	15	18	21	24	27	30	32	33

Ready kuyruğu:

PID	Bitiş zamanı	Devir süresi	Bekleme süresi
P1	32	$32 - 5 = 27$	$27 - 5 = 22$
P2	27	$27 - 4 = 23$	$23 - 6 = 17$
P3	33	$33 - 3 = 30$	$30 - 7 = 23$
P4	30	$30 - 1 = 29$	$29 - 9 = 20$
P5	6	$6 - 2 = 4$	$4 - 2 = 2$
P6	21	$21 - 6 = 15$	$15 - 3 = 12$

Ortalama devir süresi = $128 / 6 = 21,33$

Ortalama bekleme süresi = $96 / 6 = 16$

SJF (En kısa iş en önce)

Avantajlar:

- Kesintili versiyonu (SRTF) en iyi ortalama bekleme süresini verir
- Diğer algoritmalar için standart oluşturur

Dezavantajlar:

- Pratik olarak uygulanamaz çünkü süreçlerin çalışma süreleri bilinemez. Sadece tahmin edilebilir
- Uzun süren işlere hiç sıra gelmeyebilir (kıtlık)
- Süreçlere öncelik verilemez
- Uzun işlerin cevap süreleri büyük olur

Örn: SJF (kesintisiz)

PID	Geliş Zamanı	Çalışma Süresi
P1	3	1
P2	1	4
P3	4	2
P4	0	6
P5	2	3

P4	P1	P3	P5	P2	
0	6	7	9	12	16

PID	Bitiş zamanı	Devir süresi	Bekleme süresi
P1	7	$7 - 3 = 4$	$4 - 1 = 3$
P2	16	$16 - 1 = 15$	$15 - 4 = 11$
P3	9	$9 - 4 = 5$	$5 - 2 = 3$
P4	6	$6 - 0 = 6$	$6 - 6 = 0$
P5	12	$12 - 2 = 10$	$10 - 3 = 7$

Ortalama devir süresi = $40 / 5 = 8$

Ortalama bekleme süresi = $24 / 5 = 4,8$

Örn: SRTF (kesintili)

PID	Geliş Zamanı	Çalışma Süresi
P1	3	1
P2	1	4
P3	4	2
P4	0	6
P5	2	3

P4	P2	P1	P2	P3	P5	P4	
0	1	3	4	6	8	11	16

PID	Bitiş zamanı	Devir süresi	Bekleme süresi
P1	4	$4 - 3 = 1$	$1 - 1 = 0$
P2	6	$6 - 1 = 5$	$5 - 4 = 1$
P3	8	$8 - 4 = 4$	$4 - 2 = 2$
P4	16	$16 - 0 = 16$	$16 - 6 = 10$
P5	11	$11 - 2 = 9$	$9 - 3 = 6$

Ortalama devir süresi = $35 / 5 = 7$

Ortalama bekleme süresi = $19 / 5 = 3,8$