

No	Ad Soyad	İmza
----	----------	------

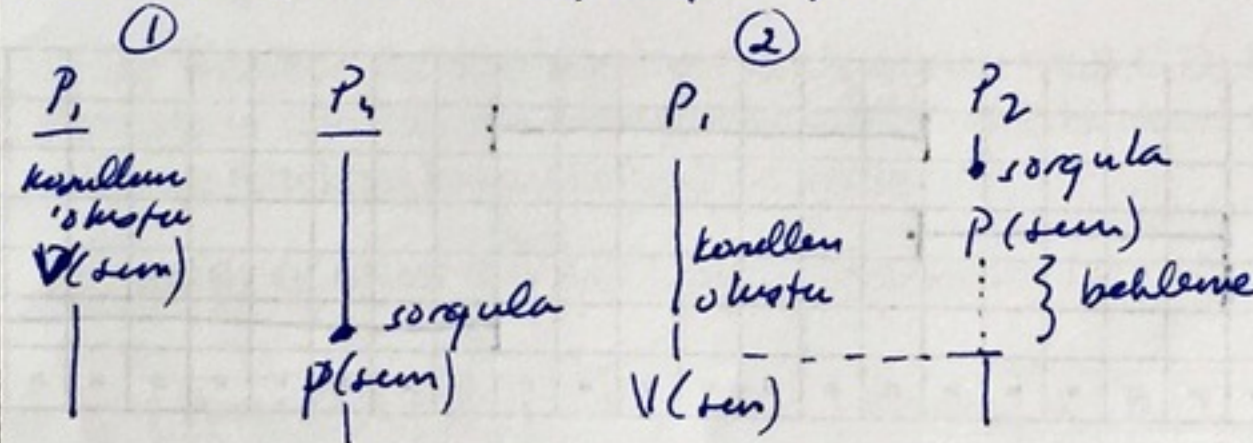
Bilgisayar İşletim Sistemleri
Arasınay II

S1	S2	S3	S4	Toplam
----	----	----	----	--------

Soru 1: (30 puan) Aşağıda verilen soruları 2-3 cümle ile yanıtlayınız.

a) Prosesler arasındaki senkronizasyon problemi nedir? Çözüm olarak semafor yapısı nasıl kullanılır?

Bir P_1 prosesinin ilerleyişi, bir diğer P_2 prosesinin yaratıcısı koşullarına bağlı ise, senk. problemi vardır. Sorgulama anında henüz kullanılmamış ise, sorgulayan proses bekler.



b) "Önce Gelen Önce Çalışır", "Dönüşümlü Sıralama" ve "Çok Düzeyli Kuyruklar" sıralama algoritmalarını kısa işlere karşı tutumları noktasında karşılaştırın.

- 10/
- Önce gelen önce çalışır \rightarrow işlerin geldiği sırayla çalışır, hata almaz, uzun işlerden sonra gelen kısa işler, çok fazla bekler.
 - Dönüşümlü Sıralama \rightarrow Her işe eşit fırsat tanır (eşit işlemci zamanı), böylece uzun süreli işlerin kısa işleri sistemde daha erken tutulabilmesi.
 - Çok düzeyli kuyruklar - Dönüşümlü sıralamaya benzer, ancak kısa işler fazla alt kuyruklara dağılır, alt kuyruğa diğer işler daha erken girer beklemesi durumunda kalabilir.

c) Paylaşılan bellek alanı yaratıp kullanmak için yararlanılan Unix sistem çağrıları nelerdir, ne yaparlar, hangi parametreleri alırlar ve bu parametreler ne ifade eder?

10/

① $\text{shmget}(\text{key} + \text{key}, \text{size} + \text{size}, \text{shmflag}) \Rightarrow$ shm id getiren
 $\text{key} \rightarrow$ paylaşılan belleğe atanacak anahtar değeri
 $\text{size} \rightarrow$ boyut bilgisi
 $\text{shmflag} \rightarrow$ erişim izni, yaratılman izni gerekli kontrol değeri

② $\text{void} * \text{shmat}(\text{int} \text{shm id}, \text{const void} * \text{shmaddr}, \text{int} \text{shmflag})$

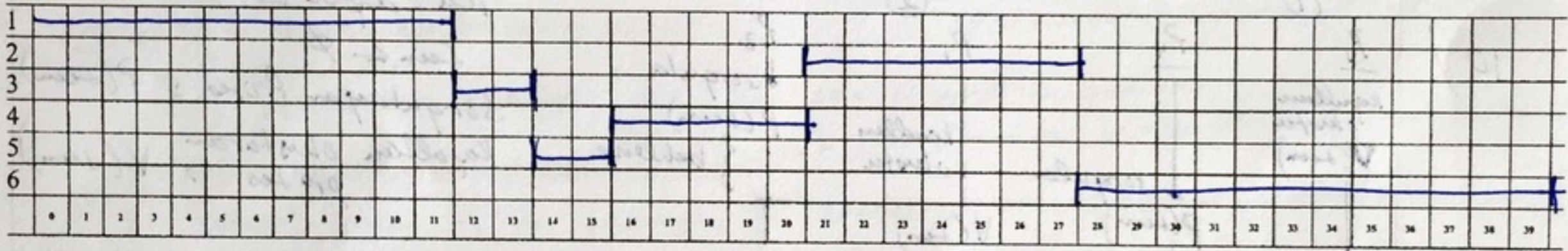
Örnek kullanım: $\text{key} = 25$, $\text{char} * \text{shm}$,
 $\text{int} \text{shm id} = \text{shmget}(\text{key}, 100, \text{IPC_CREAT} | 0666)$
 $\text{char} * \text{shm} = \text{shmat}(\text{shm id}, \text{NULL}, 0)$

*shm = "A" ---

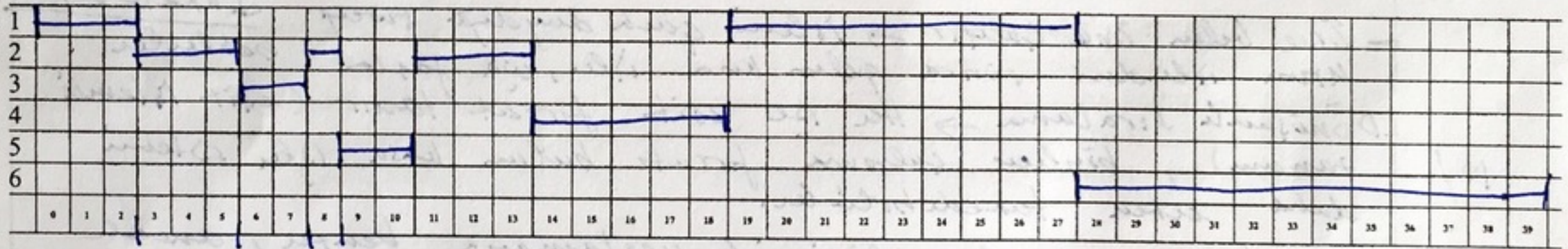
Soru 2: (20 puan) Bir bilgisayar sisteminde yer alan P1- P6, proseslerine ilişkin sisteme giriş ve gereksinim duyulan toplam işlemci zamanı bilgileri aşağıda verilmiştir. Aşağıdaki sıralama yöntemleri için proseslerin çalışmalarını gösteren zaman diyagramlarını çizin.

Proses	Sisteme Giriş zamanı	MİB Zamanı
P1	0	12
P2	3	7
P3	6	2
P4	8	5
P5	9	2
P6	12	12

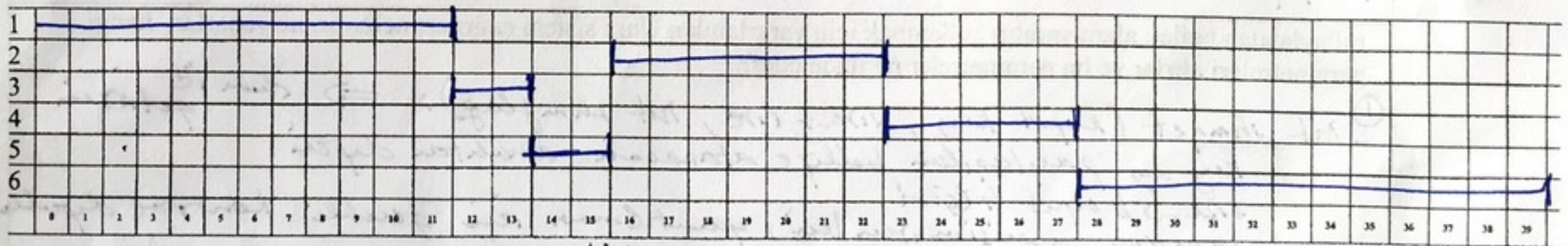
i. En kısa iş ilk önce:



ii. Kalan süresi en kısa olana göre sıralama:



iii. Yanıt süresi oranı en yüksek olana göre sıralama:



ii t Prosesler

0 — P₁-12

3 — P₁-9, P₂-7

6 — P₁-9, P₂-4, P₃-2

8 — P₁-9, P₂-4, P₄-5

9 — P₁-9, P₂-3, P₄-5, P₅-2

11 — P₁-9, P₂-3, P₄-5

12 — P₁-9, P₂-2, P₄-5, P₆-12

14 — P₁-9, P₄-5, P₆-12

19 — P₁-9, P₆-12

28 — P₆-12

iii

t₁₂

P₂ - (9+7)/7 = 2.29

✓ P₃ - (6+2)/2 = 4

P₄ - (8+5)/5 = 1.8

P₅ - (9+2)/2 = 2.5

P₆ - (0+12)/12 = 1

t₁₄

P₂ - (11+7)/7 = 2.57

P₄ - (6+5)/5 = 2.2

✓ P₅ - (5+2)/2 = 3.5

P₆ - (2+12)/12 = 1.17

t₁₆

✓ P₂ - (13+7)/7 = 2.86

P₄ - (8+5)/5 = 2.6

P₆ - (4+12)/12 = 1.3

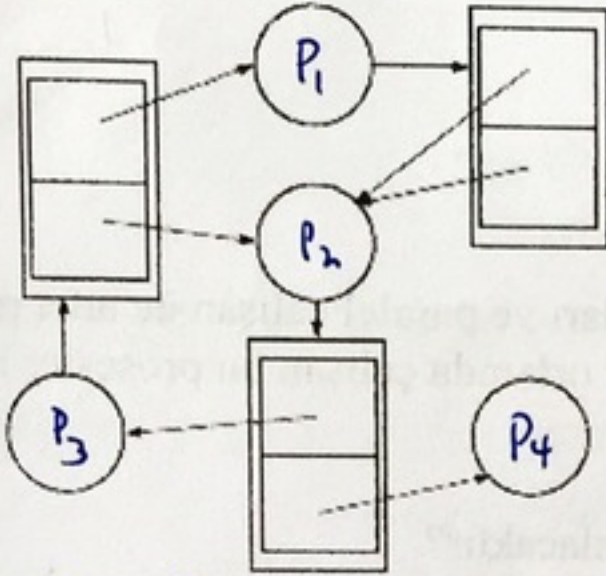
t₂₃

✓ P₄ - (15+5)/5 = 4

P₆ - (11+12)/12 = 1.92

Soru 3: (25 puan)

a)



$$P_4 - P_2 \leq P_1 - P_3$$

Yanda verilen kaynak atama şemasında daireler prosesleri, kareler de kaynakları temsil etmektedir. Bitişik kareler aynı türden kaynakları ifade etmektedir. Sistemde bir ölümcül kilitlenmenin oluşma olasılığı var mıdır?

Not: Sadece gerekçesi verilen yanıtlar değerlendirmeye alınacaktır.

Hayır - P_4 kaynak ihtiyacı yok, P_1 ve P_2 sonlanır. P_3 ise P_1 ve P_2 kaynakları kullanır, sonlanır. P_1 ve P_3 , P_2 'nin birarade kaynakları kullanıp sonlanırlar.

b) Bir sistemde yer alan dört adet farklı kaynak (A,B,C,D) beş adet proses tarafından paylaşılmaktadır. Proseslerin tüm çalışma boyunca istekte bulunabilecekleri toplam kaynak sayıları ve belirli bir anda ellerinde tuttıkları kaynaklar aşağıda verilmiştir.

SAHİP OLUNAN KAYNAKLAR

MAXIMUM İSTEK

KALAN BOŞ KAYNAK SAYISI

Proses	A	B	C	D
P0	0	0	1	2
P1	1	0	0	0
P2	1	3	5	4
P3	0	6	3	2
P4	0	0	1	4

Proses	A	B	C	D
P0	0	0	1	2
P1	1	7	5	0
P2	2	3	5	6
P3	0	6	5	2
P4	0	6	5	6

A	B	C	D
1	5	2	0

i) Kalan İstek Matrisini hesaplayınız.

ii) Başlangıçta var olan kaynak sayısını hesaplayınız

4/

Proses	A	B	C	D
P0	0	0	0	0
P1	0	7	5	0
P2	1	0	0	2
P3	0	0	2	0
P4	0	6	4	2

1/

A	B	C	D
3	14	12	12

iii) Sistemin içinde bulunduğu bu durum güvenli midir? Güvenli ise, tüm proseslerin sonlanmasını sağlayacak bir örnek kaynak atama sekansı verin.

10/

Evet, $P_0 - P_2 - P_3 \leq P_4$
veya $P_0 - P_3 - P_2 \leq P_4$

iv) Sistem yukarıda görülen durumda iken, P1 prosesi [(A,B,C,D):(0,4,0,2)] kaynakları için istekte bulunursa, Banker Algoritması'nın üreteceği yanıt ne olur, açıklayın? Yanıtınız olumlu ise, tüm proseslerin sonlanmasını sağlayacak bir örnek kaynak atama sekansı verin.

5/

Hayır, P_1 istek karşılanamaz çünkü P_1 max-istek beyan ile P_1 için D kaynağından istekte bulunamaz

Soru 4: (10 puan)

Proses 1

```
while (1) {
P(S);
putchar('A');
V(U); }
```

Proses 2

```
while (1) {
P(U);
putchar('B');
putchar('C');
V(U); }
```

Proses 3

```
while (1) {
P(U);
putchar('D'); }
```

Semaphore S=2, U=0;

Yukarıda belirtilen şekilde, ilk değerleri verilen S ve U semaforları ve paralel çalışan üç adet proses sistemde yer almaktadır. Kesintili iş sıralamanın uygulandığı bir ortamda çalışan bu prosesler için aşağıda yer alan soruları yanıtlayın.

2.0/

a) "A" kaç kere basılacaktır?

2 kere -

sem S \leftarrow 2 atarmış
ve V(S) hiç yok.

2.0/

b) "D" kaç kere basılacaktır?

- 2 kere -

- P₁, V(U) adanmış 2 kere. Yöntemleri -
sem (U) \leftarrow 2
P₂'nin P(U) - V(U) işlemi çifti birbirini
götürür.

2.0/

c) "B" en az kaç kere basılabilir, neden?

- 0 (sıfır) kere.

- P₃, P₁'den önce çalışıp, P(U) ile

sem U \leftarrow 0'a geçerse,
P₂, U üzerinde bloke olur, çalışamaz,

3/

d) Çıkış katarı "D" ile başlayabilir mi, neden?

Hayır, önce P₁'in çalışıp, "A" bastırıp, V(U) işlemi yürütmesi
gerekir ki, P₃ P(U) üzerinde askıya alınmadan, "D" bastırabilsin

Soru 5: (15 puan)

İki proses **send** (mesaj gönder) ve **receive** (mesaj al) fonksiyon çağrıları ile birbirleriyle mesajlaşarak haberleşecektir. Fonksiyonları gerçeklemek üzere, mesajı içerecek **tek gözlü bir tampon alan** kullanılacaktır. Gönderici proses, mesajı alıcı proses tarafından okunana kadar bekleyecektir. Bu yapıyı gerçekleştirme üzere aşağıda ilk değerleri verilmiş olan üç adet ikili semafor kullanılacaktır. Her semaforun ne amaç için kullanılacağını açıklayın ve **send** ve **receive** fonksiyonlarına ait kodu yazın.

Semafor boş = 1, dolu = 0, tamam = 0;

mesaj: tip

```
function send (msg: tip)
begin
P(bos);
mesaj := msg;
V(dolu);
P(tamam);
end;
```

```
function receive ()
begin
P(dolu);
msg := mesaj;
V(bos);
V(tamam);
end;
```