



**SAKARYA**  
**ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ**  
**FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ**  
**BÖLÜMÜ İŞLETİM SİSTEMLERİ DERSİ**  
**PROJE RAPORU**

**Öğrenci Bilgileri:**

- Hakan Yılmaz G181210382
- Arif Yeşilçimen G201210553
- Mehmet Ataş G201210382
- 2. Öğretim B Grubu

**Proje Adı:** GÖREVLENDİRİCİ (DISPATCHER) KABUĞU

**Proje Github Linki:** <https://github.com/hakanyilmaz96/isletimodev>

## GÖREVLENDİRİCİ TARAFINDAN KULLANILAN YAPILARIN TANIMI VE TARTIŞILMASI

Dört seviyeli önceliği olan bir görevlendirici, prosesleri işlemek için dört farklı öncelik seviyesi kullanır. Bu proseslerin öncelik seviyeleri, onların önceliklerine göre sıralanmış kuyruklarda saklanır. Önceliklerine göre sıralanan bu kuyruklardan en yüksek önceliğe sahip olan kuyrukta saklanan prosesler önce işlenir.

Gerçek zamanlı prosesler, diğer proseslerden daha yüksek bir önceliğe sahip olacak şekilde önceliksiz olarak işlenir. Gerçek zamanlı prosesler tamamlanana kadar kesilmeden yürütülür.

Normal kullanıcı prosesleri, üç seviyeli bir geri beslemeli görevlendiricide işlenir. Bu görevlendirici, "Gerçek Zamanlı" ve "Kullanıcı Proses" adı verilen iki kuyrukta saklanır. Proses listesi, her zaman adımında sürekli işlenir ve gelen prosesler uygun kuyruğa aktarılır.

Görevlendirici, prosesleri önceliklerine göre sıralanmış kuyruklarda saklar. En yüksek öncelik düzeyindeki prosesler işlenir ve daha sonra önceliği düşürülerek bir alt kuyruğa yerleştirilir. Ancak her iş öncelik değerine uygun bir kuyruğa yerleştirilir. Eğer tüm prosesler en alt seviye kuyrukta ise o zaman basit bir çevrimsel sıralı (round robin) algoritma çalıştırılır.

Tüm "hazır" yüksek öncelikli prosesler tamamlandığında, geri beslemeli görevlendirici, en yüksek öncelikli ve boş olmayan kuyruğun başındaki prosesin kaldığı yerden devam etmesiyle çalışmasını sürdürür. Bir sonraki zaman adımında, eşit veya daha yüksek önceliğe sahip başka "hazır" prosesler varsa, mevcut proses askıya alınır (veya sonlandırılır).

Bu görevlendirici, prosesleri daha verimli bir şekilde işlemek için dört öncelik seviyesi kullanır ve bu sayede proseslerin önceliklerine göre işlenmesine yardımcı olur. Gerçek zamanlı prosesler önceliksiz olarak işlenir ve normal kullanıcı prosesleri üç seviyeli bir geri beslemeli görevlendiricide işlenir. Bu görevlendirici ayrıca prosesleri önceliklerine göre sıralanmış kuyruklarda saklar ve en yüksek öncelik düzeyindeki prosesleri önce işler. Eğer tüm prosesler en alt seviye kuyrukta ise o zaman basit bir çevrimsel sıralı (round robin) algoritma çalıştırılır. Bu görevlendirici, prosesleri daha verimli bir şekilde işlemek için tasarlanmıştır.

### Olası iyileştirmeler:

Bu görevlendirici hakkında verilen bilgiye göre, görüldüğü kadarıyla bir eksiklik yok gibi görünüyor. Ancak, herhangi bir görevlendirici sisteminde, proseslerin işlenme sırasını ve önceliklerini belirleyen algoritmaların etkinliği ve verimliliği önemlidir. Bu nedenle, bu görevlendirici sisteminde proseslerin işlenme sırasını ve önceliklerini belirleyen algoritmaların etkinliği ve verimliliğini iyileştirme çalışmaları yapılması olası bir iyileştirme olabilir.

## **PROGRAMIN YAPISI ,BİREYSEL MÖDÜLLERİN TANIMI VE GEREKÇESİ**

Bu görevlendirici sisteminde, proseslerin önceliklerine göre sıralanmış dört kuyruk kullanılmıştır çünkü bu sayede proseslerin önceliklerine göre daha verimli bir şekilde işlenebilir. Önceliklerine göre sıralanmış kuyruklarda saklanan prosesler, önceliklerine göre işlenir ve bu sayede daha önemli olan prosesler önce işlenir. Böylece, önemli proseslerin tamamlanması için gereken süre daha kısa olur ve sistem daha verimli bir şekilde çalışır.

Ayrıca, gerçek zamanlı proseslerin önceliksiz olarak işlenmesi, bu proseslerin önemine göre daha verimli bir şekilde işlenmesine yardımcı olur. Gerçek zamanlı prosesler tamamlanana kadar kesilmeden yürütülür ve bu sayede bu proseslerin tamamlanması için gereken süre daha kısa olur.

Normal kullanıcı proseslerinin üç seviyeli bir geri beslemeli görevlendiricide işlenmesi ise, bu proseslerin önceliklerine göre daha verimli bir şekilde işlenmesine yardımcı olur. Görevlendirici, prosesleri önceliklerine göre sıralanmış kuyruklarda saklar ve en yüksek öncelik düzeyindeki prosesleri önce işler. Böylece, daha önemli olan prosesler önce işlenir ve daha az önemli olan prosesler daha sonra işlenir. Bu sayede, sistem daha verimli bir şekilde çalışır.

## ÖDEV HAKKINDA BİLGİ VE ÇIKTI EKLAN GÖRÜNTÜSÜ

Projemizde ilgili bütün yapılar ve sınıflar oluşturuldu. Proses sınıfı, main sınıfı içerisinde oluşturuldu. Programda FCFS algoritması sorunsuz çalışmaktadır. Fakat ödevde istenen bazı yapılar ve diğer algoritmalarındaki problemleri çözemedik ve sadece FCFS çalıştığı için aşağıdaki gibi bir çıktı almaktayız;

0 Saniye proses başladı:	PID: 0	Oncelik: 1	Kalan Zaman: 2
1 Saniye proses yurutuluyor..	PID: 0	Oncelik: 1	Kalan Zaman: 1
2 Saniye proses sonlandı:	PID: 0	Oncelik: 1	Kalan Zaman: 0
2 Saniye proses başladı:	PID: 1	Oncelik: 0	Kalan Zaman: 1
3 Saniye proses sonlandı:	PID: 1	Oncelik: 0	Kalan Zaman: 0
3 Saniye proses başladı:	PID: 2	Oncelik: 3	Kalan Zaman: 2
4 Saniye proses yurutuluyor..	PID: 2	Oncelik: 3	Kalan Zaman: 1
5 Saniye proses sonlandı:	PID: 2	Oncelik: 3	Kalan Zaman: 0
5 Saniye proses başladı:	PID: 3	Oncelik: 0	Kalan Zaman: 3
6 Saniye proses yurutuluyor..	PID: 3	Oncelik: 0	Kalan Zaman: 2
7 Saniye proses yurutuluyor..	PID: 3	Oncelik: 0	Kalan Zaman: 1
8 Saniye proses sonlandı:	PID: 3	Oncelik: 0	Kalan Zaman: 0
8 Saniye proses başladı:	PID: 4	Oncelik: 2	Kalan Zaman: 2
9 Saniye proses yurutuluyor..	PID: 4	Oncelik: 2	Kalan Zaman: 1
10 Saniye proses sonlandı:	PID: 4	Oncelik: 2	Kalan Zaman: 0
10 Saniye proses başladı:	PID: 5	Oncelik: 2	Kalan Zaman: 3
11 Saniye proses yurutuluyor..	PID: 5	Oncelik: 2	Kalan Zaman: 2
12 Saniye proses yurutuluyor..	PID: 5	Oncelik: 2	Kalan Zaman: 1
13 Saniye proses sonlandı:	PID: 5	Oncelik: 2	Kalan Zaman: 0
13 Saniye proses başladı:	PID: 6	Oncelik: 0	Kalan Zaman: 4
14 Saniye proses yurutuluyor..	PID: 6	Oncelik: 0	Kalan Zaman: 3
15 Saniye proses yurutuluyor..	PID: 6	Oncelik: 0	Kalan Zaman: 2
16 Saniye proses yurutuluyor..	PID: 6	Oncelik: 0	Kalan Zaman: 1
17 Saniye proses sonlandı:	PID: 6	Oncelik: 0	Kalan Zaman: 0
17 Saniye proses başladı:	PID: 7	Oncelik: 0	Kalan Zaman: 4
18 Saniye proses yurutuluyor..	PID: 7	Oncelik: 0	Kalan Zaman: 3
19 Saniye proses yurutuluyor..	PID: 7	Oncelik: 0	Kalan Zaman: 2
20 Saniye proses yurutuluyor..	PID: 7	Oncelik: 0	Kalan Zaman: 1
21 Saniye proses sonlandı:	PID: 7	Oncelik: 0	Kalan Zaman: 0
21 Saniye proses başladı:	PID: 8	Oncelik: 0	Kalan Zaman: 2
22 Saniye proses yurutuluyor..	PID: 8	Oncelik: 0	Kalan Zaman: 1
23 Saniye proses sonlandı:	PID: 8	Oncelik: 0	Kalan Zaman: 0
23 Saniye proses başladı:	PID: 9	Oncelik: 2	Kalan Zaman: 4
24 Saniye proses yurutuluyor..	PID: 9	Oncelik: 2	Kalan Zaman: 3
25 Saniye proses yurutuluyor..	PID: 9	Oncelik: 2	Kalan Zaman: 2
26 Saniye proses yurutuluyor..	PID: 9	Oncelik: 2	Kalan Zaman: 1
27 Saniye proses sonlandı:	PID: 9	Oncelik: 2	Kalan Zaman: 0
27 Saniye proses başladı:	PID: 10	Oncelik: 0	Kalan Zaman: 3
28 Saniye proses yurutuluyor..	PID: 10	Oncelik: 0	Kalan Zaman: 2
29 Saniye proses yurutuluyor..	PID: 10	Oncelik: 0	Kalan Zaman: 1
30 Saniye proses sonlandı:	PID: 10	Oncelik: 0	Kalan Zaman: 0