

## İŞLETİM SİSTEMLERİ DERSİ PROJE ÖDEVİ

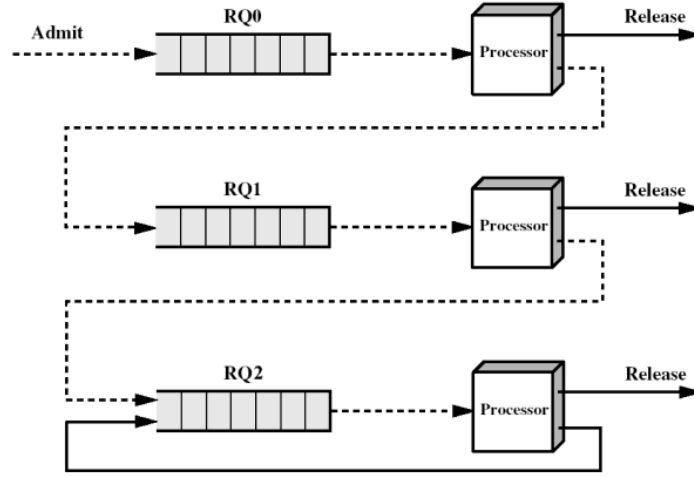
### GÖREVLENDİRİCİ (DISPATCHER) KABUĞU

Projeye konu olan sistem sınırlı kullanılabilir kaynakların kısıtlamaları içinde çalışan dört seviyeli öncelikli proses görevlendiricisine sahip bir çoklu programlama sistemidir.

#### DÖRT SEVİYELİ ÖNCELİKLİ GÖREVLENDİRİCİ

Dağıtıcı dört öncelik düzeyinde çalışır:

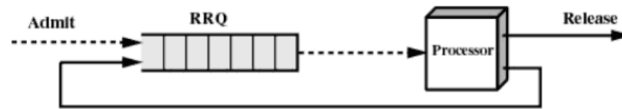
1. İlk Gelen İlk Çalışır (FCFS) algoritması temelinde hemen çalıştırılması gereken gerçek zamanlı prosesler, daha düşük öncelikte çalışan diğer proseslerden daha yüksek bir önceliğe sahiptir (öncelik değeri 0). Bu prosesler tamamlanıncaya kadar kesilmeden yürütülür.
2. Normal kullanıcı prosesleri, üç seviyeli bir geri beslemeli görevlendiricide çalıştırılır. Dağıtıcının temel zamanlama kuantumu ( $q$ ) 1 saniyedir. Bu aynı zamanda geri besleme sıralayıcısının zaman kuantumunun da değeridir.



Şekil 1. Üç Seviyeli Geri Beslemeli Görevlendirici

Görevlendirici, proses listesinden (giriş.txt) beslenen iki adet kuyruğa sahiptir: Gerçek Zamanlı ve Kullanıcı Proses kuyrukları. Proses listesi, her zaman adımında sürekli işlenir ve gelen prosesler uygun kuyruğa aktarılır. Kuyruklar işleme alınır; tüm gerçek zamanlı prosesler tamamlanmak üzere çalıştırılır ve o anda çalışmakta olan diğer düşük öncelikli prosesler kesilir.

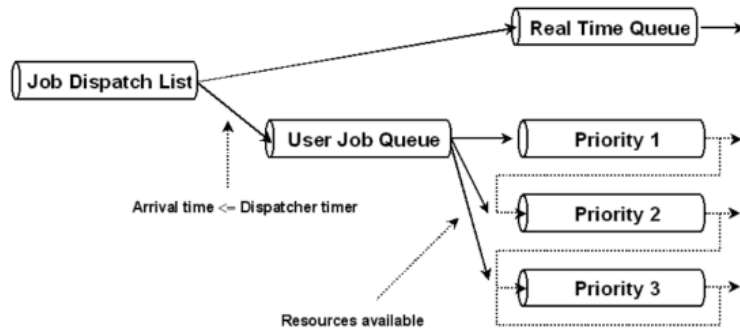
Düşük öncelikli geri beslemeli görevlendirici yeniden etkinleştirilmeden önce gerçek zamanlı kuyruktaki prosesler bitirilmelidir. Bir geri besleme kuyruğunun normal çalışması, en yüksek öncelik düzeyindeki prosesleri alır, uygun kuantuma göre işler ve daha sonra önceliğini düşürerek bir alt kuyruğa yerleştirir. Ancak her iş öncelik değerine uygun bir kuyruğa yerleştirilir (bkz Şekil 3). Eğer tüm prosesler en alt seviye kuyrukta ise o zaman basit bir çevrimsel sıralı (round robin) algoritma çalıştırılır (Şekil 2).



Şekil 2. Round Robin Görevlendiricisi

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). Mantık akışı Şekil 3'te verilmiştir.



Şekil 3. Görevlendirici Mantık Akışı

## PROSESLER



Sistem üzerindeki prosesler, sevk edilen her iş için yeni bir proses oluşturan Görevlendirici tarafından simüle edilir. Bu proses, herhangi bir öncelikli proses için kullanılabilen genel bir procestir. Prosesin fonksiyonları aşağıda listelenmiştir:

1. Proses başladığında proses kimliğini gösteren bir mesaj;
2. Prosesin yürütüldüğü her saniye düzenli bir mesaj; ve
3. Proses Askıya Alındığında, Devam Edildiğinde veya Sonlandırıldığında bir mesaj.

Görevlendirici tarafından sonlandırılmazsa, proses 20 saniye sonra kendiliğinden sona erecektir. Proses, her benzersiz proses için rastgele oluşturulmuş bir renk şeması kullanarak yazdırır, böylece proseslerin tek tek "dilimleri" kolayca ayırt edilebilir.

Bir prosesin yaşam döngüsü:

1. Proses, varış zamanı, önceliği, gereken proses süresini (saniye cinsinden) ilk proses listesi aracılığıyla Görevlendirici giriş kuyruklarına gönderilir.
2. Bir proses "vardığında" çalışmaya hazırdır.
3. Bekleyen Gerçek Zamanlı prosesler, FCFS esasına göre yürütülmek üzere gönderilir.
4. Daha düşük öncelikli bir kullanıcı prosesi, proses geri besleme Görevlendirici birimi içindeki uygun öncelik sırasına aktarılır.
5. Bir proses başlatıldığında Görevlendirici proses parametrelerini Kimliği, öncelik, kalan proses süresi (saniye olarak) görüntüler.
6. Bir Gerçek Zamanlı prosesin, zaman süresi dolana kadar çalışmasına izin verilir.
7. Düşük öncelikli bir Kullanıcı prosesinin, askıya alınmadan veya süresi dolmuşsa sonlandırılmadan önce bir saniye boyunca çalışmasına izin verilir. Askıya alınırsa, öncelik seviyesi düşürülür (mümkünse) ve yukarıdaki Şekil 1 ve 3'te gösterildiği gibi uygun öncelik kuyruğunda yeniden kuyruğa alınır.
8. Gelen prosesler kuyruğunda daha yüksek öncelikli Gerçek Zamanlı prosesler beklemediği sürece, geri besleme kuyruklarındaki en yüksek öncelikli bekleyen proses alınır veya yeniden başlatılır.
9. Proses listesinde, giriş kuyruklarında ve geri besleme kuyruklarında başka proses olmadığında, Görevlendirici sona erer.

## PROSES LİSTESİ

Proses Listesi, işlenecek proseslerin listesidir. Liste, komut satırında belirtilen bir metin dosyasında bulunur. Yani

>systemd giriş.txt

Listenin her satırı, aşağıdaki verilerle "virgül" ile sınırlandırılmış liste olarak bir prosesi tanımlar:

<varış zamanı>, <öncelik>, <proses zamanı>

Mesela,

12, 0, 1

12, 1, 2

13, 3, 6

şunu belirtir:

1. Proses: 12 zamanında varış, öncelik 0 (Gerçek Zamanlı), 1 saniyelik proses zamanı
2. Proses: 12 zamanında varış, öncelik 1 (yüksek öncelikli Kullanıcı prosesi), 2 saniye proses süresi,
3. Proses: 13 zamanında varış, öncelik 3 (en düşük öncelikli Kullanıcı prosesi), 6 saniyelik proses süresi,

İş/proses listesini içeren metin dosyası projede verilecektir.

## PROJE GEREKSİNİMLERİ

1. Yukarıdaki kriterleri karşılayan bir Görevlendirici tasarlayın.

Rapor içerisinde:

- a. Çeşitli modülleri ve ana işlevleri açıklayarak programınızın genel yapısını tanımlayın ve gerekçelendirin ('arayüzler' işlevinin açıklamaları beklenir) .
- b. "Gerçek" işletim sistemleri tarafından kullanılan şemalarla karşılaştırarak, böyle çok düzeyli bir görevlendirici şemasının neden kullanılacağını tartışın. Olası iyileştirmeler önererek, böyle bir şemadaki eksiklikleri ana hatlarıyla belirtin. Tartışmalarınıza bellek ve kaynak ayırma şemalarını dahil edin.

Raporun derinlemesine tartışmalara, açıklamalara ve argümanlara sahip olması beklenir.

**Rapor herhangi bir kaynak kodu İÇERMEMELİDİR. Pdf olarak sisteme yüklenecek ve tüm grup üyelerinin adlarını içeren bir kapağa sahip olacaktır.**

2. Dağıtıcıyı JAVA dilini kullanarak kodlayın.
3. Kaynak kodu, meslektaşlarınızın kodu anlamasını ve kolayca değerlendirmesini sağlamak için kapsamlı bir şekilde yorumlanmalı (açıklama satırları - Türkçe) ve uygun şekilde yapılandırılmalıdır. Düzgün yazılmış ve düzenlenmiş kodu yorumlamak çok daha kolaydır ve projenizi değerlendirecek öğretim elemanının zihinsel jimnastik yapmak zorunda kalmadan kodlamanızı anlayabilmesini sağlamak sizin yararınıza!
4. SABİS'e yüklenecek zip dosya Eclipse derlenmiş projesini ve pdf halindeki raporu içermelidir.
5. Tüm proje notu sınıf notunuzun %15'i değerindedir. Kopya ödevler 0 (sıfır) ile notlandırılacaktır. Ödevler GitHub'a grup temsilcisi tarafından yüklenecektir. Yüklenen proje dosyasının adı grup adı olacaktır. Ders yürütücülerine ve ders yardımcılara GitHub üzerinde proje dosyalarına erişim yetkisi (sadece okuma) verilecektir. Grubun diğer üyeleri en az bir commit gerçekleştirmelidir. Commit yapmayan üyeler 0 olarak değerlendirilecektir.

**PUANLAMA**

- Görevlendirici tarafından kullanılan yapıların tanımı ve tartışılması-10 puan
- Program yapısının ve bireysel modüllerin tanımı ve gerekçesi-10 puan
- Görevlendiricinin tartışılması eksiklikler ve olası iyileştirmeler - 10 puan.
- FCFS yüksek öncelikli sıralayıcının çalışması - 10 puan
- Kullanıcı Geri Beslemeli sıralayıcının çalışması - 10 puan
- Kullanıcı Geri Beslemeli sıralayıcının Round Robin modunda çalışması - 10 puan
- Karışık sıralayıcı çalışması - 10 puan
- Kuyruk sınıfı – 10 puan
- Proses sınıf – 10 puan
- Ana program - konsol uygulaması– 10 puan