

Windows İşletim Sistemi Üzerinde FIFO Proses Yönetimi İçin Yazılım Geliştirilmesi

Onur ÜLGEN¹

Mutlu AVCI²

Çukurova Üniversitesi
Mühendislik ve Mimarlık Fakültesi
Bilgisayar Mühendisliği Bölümü
Balcalı / ADANA

¹e-posta: onurulgen@gmail.com

²e-posta: mavci@cu.edu.tr

Anahtar Sözcükler: FIFO, Tek Yönlü Bağlı Liste, İş Sıralama, Proses Yönetimi

Özet

Bu çalışmada FIFO iş sıralama algoritması ve bu algoritma kullanılarak bir proses yöneticisinin Windows işletim sistemi üzerinde C yazılım dili kullanılarak nasıl yazılacağı anlatılmıştır.

1. Giriş

Bütün modern bilgisayarlar aynı anda pek çok işi yapabilir gibi görünmektedirler. Genellikle Merkezi İşlemci Birimi (CPU), her prosesi onlarca veya yüzlerce mikro saniye çalıştırarak anahtarlama yapmaktadır. Tek işlemcili sistemlerde aslında bir anda sadece bir işlem yapılmaktadır. Bu hızlı anahtarlama sözde bir paralelliktir. Gerçek bir paralellik ancak iki veya daha fazla işlemci içeren sistemlerde mümkündür. İşletim sistemi tasarımcıları paralellikle kolayca uğraşabilmek için sıralı işlemler modelleri geliştirmişlerdir [1].

Her proses program sayıcısı ve iç durumlarıyla bağımsız olmasına rağmen başka proseslerle ilişkiye sık sık ihtiyaç duyar. Bir proses başka bir proses için giriş olacak çıkışlar üretebilir veya bir girişi bekleyebilir. Bu nedenle her bir proses durum modeline göre; hazır, bloklamış veya çalışır hallerden birinde bulunur [2].

Tüm çalıştırılabilir yazılımlar, işletim sistemi de dâhil olmak üzere sıralı prosesler olarak organize edilir. Bir proses sadece program sayıcı, yazmaç ve değişkenlerin o anki değerlerini içererek çalışan programdır. Her proses kendi zahiri CPU'suna sahip gibidir. Gerçek CPU aslında prosesler arasında ileri geri anahtarlama yapmaktadır. CPU'nun prosesler arası paylaşılmasına iş sıralama adı verilir [3].

İş sıralamasında göz önüne alınması gerekenler; dengeli olması yani tüm proseslerin aynı etkilenmesi ve sonsuz erteleme olmaması, birim zamanda maksimum prosese izin verilmesi, etkileşimli

kullanıcıların yanıt süresinin en aza indirilmesi, kaynak kullanımının dengelenmesi, yanıt süresi ile kaynak kullanımı arasında denge sağlanmasıdır.

İş sıralamada öncelik mekanizması olmalı, sonsuz bekleme önlenmeli, kilit kaynakları kullanan prosesler hemen çalıştırılıp kaynakları bırakmaları sağlanmalı ve çok yüklü durumlarda sistemin çökmesi önlenmelidir.

İş sıralama kriterleri basitçe prosesin giriş/çıkış eğilimi, CPU kullanma eğilimi, etkileşimli yada batch mi olduğu, yanıt alma süresinin acilliği, önceliği, ne sıklıkla sayfa hatası ürettiği, daha yüksek öncelikli proses tarafından yarıda kesilme sıklığı, ne kadar gerçek yürütme zamanı elde ettiği, sonlanmak için ne kadar zamanı kaldığı gibi faktörlerdir.

İlk Giren İlk Çıkar (FIFO) algoritması ise paralel işlem yapamayan bilgisayarların kullandığı bir iş sıralama algoritmasıdır. Bu yapıya gelen bir proses kuyruğun en sonuna geçer. Sıra kendine gelene kadar bekler. Sırası gelen proses işlenir ve bu proses sonlanana kadar başka proses çalıştırılmaz.

Bu çalışmada amaç, Windows işletim sisteminde var olan proses yönetimini kullanmak yerine, çalıştırılacak olan proseslerin bu çalışmada geliştirilen FIFO iş sıralama algoritmasına göre yönetilmesini sağlamaktır.

Bu işlemi gerçekleştirmek için C yazılım dilinde kod geliştirilmiş, Windows işletim sistemi üzerinde çalıştırılarak FIFO iş yönetimi gerçekleştirilmiştir.

2. FIFO İş Sıralama Algoritması

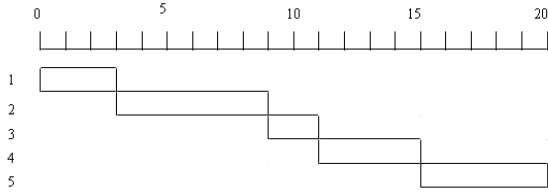
FIFO en basit, kesintisiz iş sıralama algoritmasıdır. Sisteme geliş sıralarına göre prosesler tamamlanana kadar işlenir. Uzun prosesler yüzünden kısa prosesler beklemede kalırlar. Giriş/çıkış işleminin yoğun olduğu prosesler için pek iyi çözüm değildirler. Ancak aynı önceliğe sahip prosesler için uygulanabilirler.

Tablo 1’de beş adet farklı proses geliş anları ve hizmet süreleri görülmektedir.

Proses	Geliş Anı (sn)	Hizmet Süresi (sn)
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2

Tablo 1. Prosesler, Geliş Anları ve Hizmet Süreleri

Şekil 1’de Tablo 1’de verilen beş adet proses için FIFO iş sıralama algoritması ile elde edilecek zaman diyagramı verilmiştir.



Şekil 1. FIFO Proses-Zaman İş Yönetim Diyagramı

Şekil 1’de görüldüğü gibi bir proses tamamlanmadan sonraki processe geçilmemektedir. FIFO kesintisiz bir kuyruk yapısıdır.

FIFO Kuyruk yapısını oluşturmak için genellikle Tek Yönlü Bağlı Listeler kullanılır.



Şekil 2. FIFO Tek Yönlü Bağlı Liste Yapısı

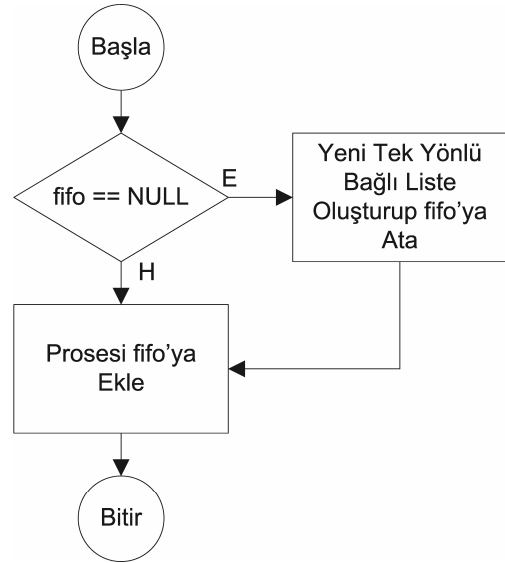
Listenin yapısı Şekil 2’deki gibidir. Yapının veri olarak adlandırılan ilk elemanı proses adresini tutar. İkinci eleman ise sonraki prosesin adresini saklayan bağlantılı liste elemanının adresini saklayan işaretçidir.

3. FIFO Proses Yönetiminin Gerçekleştirilmesi

3.1 FIFO Proses Yöneticisine Proses Ekleme

Windows’a bir proses eklemek istendiğinde CreateProcess Uygulama Programlama Arayüzü (API) çağrılır. Bu API Windows’un proses yöneticisinin kuyruğuna yeni bir proses ekler. Eğer bu API olmasaydı Windows’ta program çalıştırılmazdı.

FIFO Proses Yöneticisi de benzer bir mantıkla tasarlanmıştır. Yöneticiye proses eklenmesi için bir API gereklidir yoksa yönetici hiçbir iş yapmadan bekleyecektir.



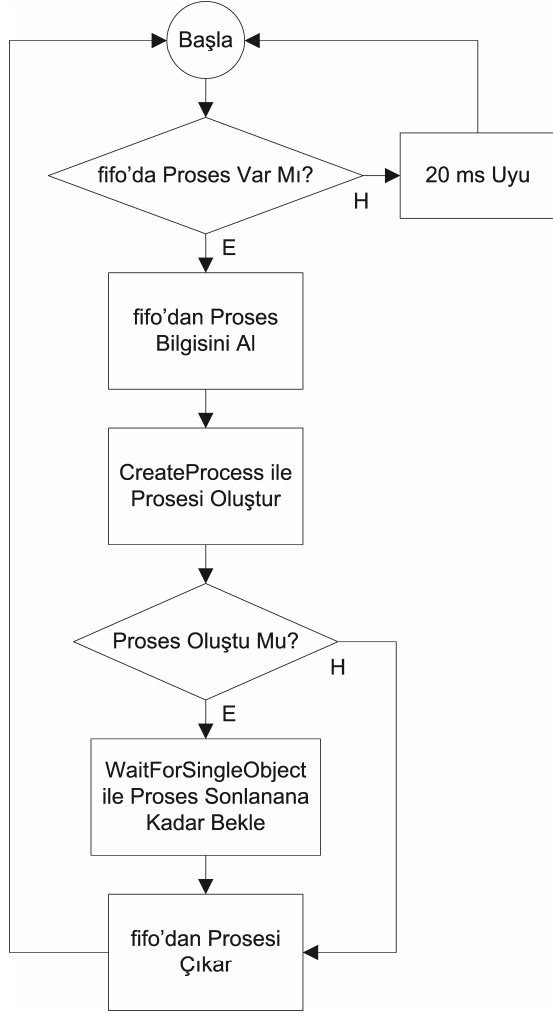
Şema 1. Proses Ekleme Fonksiyonunun Akış Şeması

Şema 1’deki FIFO Proses Yöneticisi’nin Proses Ekleme fonksiyonu ilk önce *fifo* değişkeninin NULL olup olmadığını kontrol eder. Eğer NULL ise yeni bir Tek Yönlü Bağlı Liste oluşturup *fifo* değişkenine atar.

FIFO Proses Yöneticisi procesi kendi oluşturacağı için fonksiyon parametre olarak prosesin disk üzerinde nerede tutulduğu bilgisini alır ve aldığı bu veriyi *fifo* kuyruğuna ekler.

3.2 FIFO Proses Yöneticisi Ana Programı

FIFO Proses Yöneticisinin görevi FIFO kuyruğunda sırası gelen procesi, sonlanana kadar çalıştırmaktır. Buna kuyruқта proses kalmayınca kadar devam eder. Kuyruқтаki prosesler bitince de kuyruğa bir proses gelene kadar uyur.



Şema 2. Ana Program Akış Şeması

Program bir sonsuz döngüde çalışmaktadır. Sürekli hafızada kalıp gelen isteklere cevap verebilmesi için böyle tasarlanmıştır.

Program, eğer *fifo*'da proses yok ise 20 ms uyuyup en başa dönmektedir. 20 ms beklemesinin sebebi bu sürenin Windows'un kuantum süresi olmasıdır. Yani Windows bu süre dolunca çalıştırdığı ipliği durdurup diğer ipliğe geçiş yapar. Hem işlemciyi meşgul etmeden hem de eklenen prosesi fazla bekletmeden çalıştırabilmek için bu süre seçilmiştir.

Program, `CreateProcess` fonksiyonunu çağırarak prosesi oluşturmaya çalışır. Başarılı olup prosesi oluşturabilirse `WaitForSingleObject` fonksiyonuyla proses sonlanana kadar bekler.

Program son olarak prosesi kuyruktan çıkarır ve en başa dönerek bu işlemleri kuyruktaki diğer elemanlar için tekrarlar.

4. Sonuç

FIFO Proses Yöneticisi batch proses çalıştırma işlemlerinde kullanılabilir. Örneğin; FIFO Proses Yöneticisi kullanılarak Windows'taki BAT uzantılı dosyaları çalıştırılabilir. Ayrıca FIFO Proses Yöneticisi esnek zaman paylaşım yapıya getirilerek Windows işletim sisteminin proses yöneticisinin yaptığı işi yapabilir.

Bu çalışmada FIFO İş Sıralama algoritması kullanılarak proseslerin nasıl yönetileceği uygulamalı olarak anlatılmış ve Windows işletim sistemi üzerinde FIFO Proses Yöneticisi yazılımının C yazılım dili ile nasıl gerçekleştirildiği açıklanmıştır.

Kaynaklar

- [1] A. Tanenbaum and A. Woodhull, "Operating Systems Design and Implementation", Prentice Hall, 2006
- [2] W. Stallings, "Operating Systems: Internals and Design Principles", Prentice Hall, 2004
- [3] A. Silberschatz, "Operating System Concepts", Wiley, 2004