

İşletim Sistemleri - Görevlendirici

Elman Muradov Mehmet Ali Demirtaş Emirhan Kaya
Ahmet Hakan Arı Ebubekir Mert

26 Aralık 2022

İÇİNDEKİLER

1	Giriş	2
1.1	Görevlendiricilerin Gerekliliği	2
2	Tasarımı	3
2.1	Görevlendirici	3
2.2	Dağıtıcı	3
2.3	Sıra	3
2.4	Proses	4
3	Karşılaştırma	5
3.1	MS-DOS	5
3.2	Klasik UNIX	5
3.3	SVR2 UNIX	5
3.4	Linux	6
3.5	Windows	6
4	Kaynakça	7

BÖLÜM 1

GİRİŞ

1.1 Görevlendiricilerin Gerekliliği

Görevlendiriciler çoklu işlem yapabilen çekirdek ve işletim sistemleri için hayati bir öneme sahiptir. Proseslere sınırlı işlemci gücünün ihtiyaçlara göre dağıtılmasını üstlenen görevlendiriciler modern işletim sistemlerinin işlevlerinin yerine getirilebilmesi için en önemli sistemlerden bir tanesidir.

BÖLÜM 2

TASARIMI

2.1 Görevlendirici

Burda bir araya getirdiğimiz görevlendirici SVR2 UNIX ve Linux'un eski versiyonlarının kullandığına benzer bir sistemde inşa edilmiştir. $O(n)$ notasyonuna sahip; öncelik sıralamalı, çok seviyeli ve geri dönüşlü bir görevlendiricidir. Öncelik seviyesi "00" gerçek zamanlı proseslere ayrılmıştır. Çekirdek ve kullanıcı prosesleri ayrılmamıştır. Bu sebeple çekirdek proseslerinin zaman aşımı riski vardır.

2.2 Dağıtıcı

Görevlendiricinin farklı seviyelerinde kullanılan ortak dağıtıcı sistemi bulunan öncelik seviyesine göre farklı işlev göstermek için tasarlanmıştır. Gerçek zamanlı proseslerin bulunduğu "00" öncelik seviyesinde FCFS prensiplerine uygun bir şekilde hareket ederken daha düşük öncelikli prosesler geri dönüşlü olarak çalışırlar. Sadece bir öncelik seviyesinde tamamlanmamış proseslerin bulunması durumunda ise round-robin modunda işlev görür.

2.3 Sıra

Dağıtıcıların kullandığı ortak sıra implementasyonu proseslerin sıra içerisindeki hareketini yerine getirmek ve Round-robin veya gerçek zamanlı FCFS prensiplerine uyum sağlayabilmesi için farklı işlev gösterebilecek şekilde tasarlanmıştır. Bu işlevler ise kullanılan öncelik seviyesi, tamamlanmamış proseslerin dağılımı gibi faktörleri göz önüne alarak koordine eder.

2.4 Proses

Görevlendiricinin işlemlerini gerçekleştirirken prosesleri betimleyen yapılar kullanılmaktadır. Projemizde bu yapılar gerçek bir prosesin sahip olabileceği önemli değişkenleri saklar. Ayırt edici bir kimlik numarası, kalan zaman, başlangıç anı, öncelik seviyesi, yürürlükte olduğu toplam zaman gibi değişkenler görevlendirici tarafından kullanılarak işlemci zamanı paylaşılır.

BÖLÜM 3

KARŞILAŞTIRMA

3.1 MS-DOS

İlk MS-DOS sistemleri çoklu görev yapabilen çekirdekler olmadıklarından bir görevlendirici kullanmamıştır. Tek bir proses kullanıldığı için performans bakımından çoklu görev yapan sistemlere kıyasla daha yüksek performans alınmasına olanak sağlasa da aynı anda iki prosesin (çekirdek prosesleri de dahil olmak üzere) çalışmasını imkansız kılmıştır. MS-DOS, proseslerin çekirdeğe ne kadar süre çalışmaları gerektiğini bildirmesine dayanan kooperatif çoklu işlem sistemine dayanmaktadır. $O(1)$ notasyonuna sahiptir.

3.2 Klasik UNIX

Klasik UNIX sistemleri yerini alan diğer çekirdeklerin aksine öncelik sistemli bir görevlendirici kullanmamıştır. Tek seviyeli bir dağıtıcı sistemi "round-robin" modunda çalışır. FCFS sistemi kullanılır ancak sadece çekirdek prosesleri için kullanılır. Bu sistem herhangi bir öncelik sistemi kullanmadığı için proseslerin artması durumunda sıralamada altta kalan proseslerin zaman aşımına uğrama olasılığı daha yüksektir. Bizim bu projede yarattığımız görevlendirici öncelik sıralamasını kullandığı için bu sorunları yaşama olasılığı çok daha düşüktür. $O(n)$ notasyonuna sahiptir.

3.3 SVR2 UNIX

Klasik UNIX sistemlerinin aksine SVR2 UNIX çok seviyeli öncelik sistemli ve geri dönüşlü dağıtıcıları kullanan bir görevlendirici kullanmıştır. Bunun yanı sıra kullanım bazlı öncelik sistemiyle atası olan klasik UNIX sistemlerinin zaman aşımı sorununa çözüm bulmuştur. Kullanıcı proseslerine en düşük öncelikli

çekirdek proseslerinden daha düşük bir öncelik değeri atanarak çekirdek proseslerinin zaman aşımına uğramasının önüne geçilmiştir. Bizim yarattığımız görevlendirici ile karmaşıklık ve kullanım bakımından benzer bir sisteme sahiptir. $O(n)$ notasyonuna sahiptir.

3.4 Linux

Linux yaşam süresi boyunca birden fazla görevlendirici kullanmıştır. Linux versiyonu 2.4'ten versiyon 2.6'ya kadar $O(n)$ notasyonuna çok seviyeli öncelik sistemli ve geri dönüşlü bir dağıtıcı sistemini kullanmıştır. Bu versiyonların kullandıkları görevlendirici bizim görevlendiricimize tasarım anlamında yakındır ve aynı işlevleri yerine getirir. Linux versiyonu 2.6'dan versiyon 2.6.22'ya kadar $O(1)$ notasyonuna sahip bir görevlendirici kullanılmıştır. Linux versiyon 2.6.22'dan sonra ise bu sistem yerine $O(\log n)$ notasyonuna sahip "Completely Fair Scheduler" kullanılmıştır.

3.5 Windows

Windows, MS-DOS tabanlı versiyonlarında MS-DOS uygulamalarıyla uyumluluk sağlanabilmesi için işlemcinin 16-bit gerçek modunda ("Real Mode") kooperatif çoklu işlem modunda çalışmaktadır. İşlemcinin 32-bit korunumlu modunda ("Protected Mode") ise çok seviyeli, öncelik seviyeli ve geri dönüşlü bir görevlendirici kullanılmıştır. Windows NT tabanlı işletim sistemlerinde ise bu sistem kullanılmaya devam edilmiştir. $O(n)$ notasyonuna sahip bir görevlendiricidir, tasarım ve işlev olarak bizim görevlendiricimize benzerdir.

BÖLÜM 4

KAYNAKÇA

- Ken Thompson, "UNIX Implementation", 2.3 - Synchronization and Scheduling, Bell Laboratories
- Maurice J. Bach, "The Design of the UNIX Operating System", Chapter 8 - Process Scheduling and Time, Prentice Hall
- Wikipedia, "Scheduling (computing)", Process Scheduler
- Wikipedia, "Scheduling (computing)", Operating system process scheduler implementations