



OPERATING SYSTEMS PROJECT 3 REPORT

Ad	Abdussamet
Soyad	KACI
NO	1721221007
Tarih	01.01.2021

Proje Adı

Thread Senkronizasyonu

Proje Konusu

Bu projede thread senkronizasyonunu sağlamak ve bunun hakkında tecrübe edinilmesi amaçlanmıştır. Projemizde matris işlemleri yapılmaktadır ve bu matris işlemleri thread kullanılarak her bir thread'de farklı işlemler gerçekleştirilmektedir.

Thread Nedir?

Thread, bir process'de yapılması gereken küçük bir iş olarak görülebilir. İngilizce'de iplik anlamına gelen threadler, bilgisayar biliminde iş parçacığı olarak adlandırılmaktadırlar.

Bir process'de birden fazla iş parçacığı bulunabilir ve process'ler işlemcinin yapacağı işlerin gruplanmış hali olarak düşünülebilir. Process'ler işlerinin sırasını atlayamazlar ve yapması gereken işleri yerine getirmek zorundadırlar. Thread'ler process'lerin altında çalışırlar ve processlere nazaran işlemcide daha az maliyetle çalışırlar. Process altında bir thread'in çalıştırılması sayesinde process o iş parçacığını beklemeden ana işlerini yapmaya devam edebilir.

Thread'leri kullanmakla beraber birden fazla thread kullanılması durumunda eğerki ortak kullanılan bir memory alanı varsa (kritik bölge) bu alanın kontrol altında olması gerekmektedir. Çünkü threadler bir yarış halindedirler (race condition) ve işlemciye ilk hangisi varırsa o çalışır. Bundan dolayı bir değişken üzerinde sırasıyla bir işlem yapılması gerekiyorsa bu gerçekleşmeyebilir. Bunu sağlamak için de yani thread'lerin senkron çalışması için kilit mekanizmalarını kullanırız.

Proje Hakkında

Projemizde 1 adet program ve 1 adet makefile bulunmaktadır.

Programımız 20x20 boyutunda bir matris oluşturmaktadır ve oluşturulan bu matrisin elemanlarının toplamını alıp ekrana

bastırmaktadır. Bu işleri yaptırırken multithread yapmaktayız. Yani birden fazla thread kullanılmaktadır.

Projede 2 adet thread kullanılmaktadır (thread_matrix_creator ve thread_matrix_sum). thread_matrix_creator adlı thread 20x20 boyutunda matris oluşturup ekrana bastırmaktadır. thread_matrix_sum adlı thread ise oluşturulan bu matrisi toplayıp ekrana bastırmaktadır. Bu threadler bir döngü içerisinde çalışmaktadırlar.

Ayrıca bir de ana programımız bulunmaktadır. Bu programımızda kullanıcıdan kaç adet matris oluşturulacağı istenmektedir. Girilen adet sayısınca threadler işlerini tamamlarlar ve tekrar kullanıcıdan sayı istenir. Kullanıcı sıfır veya negatif sayı girene kadar bu döngü devam eder.

Ayrıca bu projede sinyal yakalama işlemi de gerçekleştirilmiştir. Kullanıcı programı kullandığı süre boyunca CTR+C veya CTRL+Z gibi tuş takımlarından herhangi birine basarsa programa bir çıkış sinyali gönderilmiş olur ve threadler sonlandırılıp programdan çıkılır.

Projede Karşılaşılan Zorluklar

Projede genel olarak threadlerde senkronizasyonu sağlarken bazı sıkıntılar yaşadım. Bir thread'i lock ve unlock yaparken yerlerini karıştırdım ve istenmeyen bazı durumlar yaşadım. Örneğin programımda bir counter tutmaktayım ve bu counter threadlerin kaç defa çalıştırılacağını tutmaktadır. Ancak counter değişkenini kilitlediğim alanların dışında (kritik bölge dışında) artırdığımda kendiliğinden fazladan artmaya başladı ve daha az sayıda matris oluşturuldu. Bunun sebebi de kritik bölgenin dışında counter'ı artırmamdan ve işlemciye istediğimden önce varması sebebiyle (race condition) fazladan artış gösterdi.

Ayrıca print işlemlerinde bir problem yaşadım. Yaşadığım problem ise thread içerisinde matrislerin toplamını bastırırken hemen toplamın basılmayıp main'deki print işleminin gerçekleşmesi ve sonrasında toplamın basılmasıydı. Bu sorunla az karşılaştım da problemi araştırdım. Araştırmalarım göre threadlerde print işleminden hemen sonra fflush(stdout) kullanılarak buffer'ın temizlenmesi yönündeydi. Ancak bu yöntem bende işe yaramadı. Bunun neden kaynaklandığını araştırdım. Bulduklarım göre print işlemini yapınca bu string bir

buffer'a atanmaktadır ve hemen unlock işlemi yapıldığında bu string ya ekranda görünecekti ya da main'e döndüğünde ordaki print işlemi çalışıyordu ve daha sonra matris toplamı ekranda görünecekti. Threadlerde print işleminden sonra main'e dönülmesinden dolayı böyle bir sorun ile karşılaştım. Ben de sorunu main'de bulunan printde önce sleep fonksiyonu ile çok kısa bekleyerek çözdüm.

Projede Yapılanlar ve Yapılmayanlar

Projede yapılmayan hiçbir iş bulunmamaktadır. Tüm istenilenler eksiksiz bir şekilde yerine getirilmiştir.

Kaynaklar

Lab videoları

Lab kodları

<https://stackoverflow.com/questions/23807546/c-pthreads-mutex-not-working-as-expected>

<https://stackoverflow.com/questions/12216367/why-doesn%C2%B4t-my-pthread-function-print-out-what-i-have-expected>

<https://stackoverflow.com/questions/8806085/is-this-a-printf-pthread-bug-or-am-i-missing-something>