

# SİSTEM PROGRAMLAMA

Konu: İş parçacığı kullanarak ;

İki dizinin kareköklerini üçüncü bir diziye toplamak.

Ad: Gamze

Soyad:Söğüt

Numara:180996

İş parçaları(thread) aynı bellek alanında çalışırlar.

Birbirleri arasında veri transferi yapmak için ek bir transfer yöntemine ihtiyaç duymazlar.

İşlemler ve iş parçacıkları arasındaki farka bakacak olursak;

-İş parçacıkları daha hafiftir.

-İş parçaları işleme aittir.

-İşlemlerde ana işlem çıktığı zaman alt işlem çalışmaya devam eder. İş parçalarında herhangi parçadan çıktığımız anda tüm iş parçaları sonlanmaktadır.

Eğer hafızayı paylaşmak istiyorsak iş parçacığı kullanmamız gerekiyor. Hızlı çalışmasını istiyorsak yine iş parçacığı kullanmamız gerekiyor.

İş parçacığında void kesinlikle kullanmamız gerekir. İş parçası fonksiyonuna eğer hangi değer göndereceksek bu parametrede belirtilmektedir.

Yeni bir iş parçacığı oluşturmak istiyorsak;

`Pthread_create(&isparcasi,nitelikler,fonksiyon,parametre)`

Kütüphanemiz ise `<pthread.h>` dır.

Derleyici olarak `-pthread` gerekmektedir.

Daha sonra iş parçalarını birleştirme işlemini yaparız. Bunu da `pthread_join(isparcasi,donendeger)` olarak yazarız. Dönen değer kısmına NULL da yazabiliriz. Bu çok önemli değildir.

Her bağlanan kişi için bir tane thread oluşturuyoruz. Sunucu da tek bir process altında birden fazla işlem yapılıyor.

Bir process altında birden fazla thread çalışabilmektedir. Sunucu tarafından sürekli dinleyen bir thread çalışıyor. Eğer bağlantı için birisi geldiği zaman onu başka bir thread e atmaktadır.

Bir process in tek başına olması yetmemektedir. Bir process altında birden fazla iş yapılması gerekebiliyor. Bu yüzden process üretildiği zaman altında thread da üretiliyor.

Thread de değişken tanımladığımızda bu değişkenin değeri değiştiğinde diğer bütün threadlerde de değişken değeri değişmektedir. Yalnız process de yeni bir process üretildikten sonra değişkenin bir kopyası yeni bir process için oluşturulur. Ram da iki ayrı değişken olur. Birinin değişmesi diğerinin değişmesi anlamına gelmez.