Bu ödevde basitleştirilmiş şekilde bir bilgisayardaki işletim sistemine gelen isteklerin yani işlemlerin nasıl sırayla kaynaklara erişip, görevlerini gerçekleştirdiğini göreceğiz.

Sizin göreviniz mutex kullanarak semafor yazmak. Bildiğiniz üzere mutex aslında semaforun tekil hali, yani 0-1 değeri gibi çalışan hali. Biz yeni bir sayac değişkeni daha tanımlayarak bu tekil yapıyı, çoğul yapıya yani semafora dönüştüreceğiz.

Ödevinizde öncelikle kuracağımız bu yapay sistemde kaç işlem çalışacağını ve işlem kuyruğuna aynı anda maksimum kaç işlemin erişeceğini kullanıcıdan alacağız. Kuyruğa erişim izni alanlar işlemlerini gerçekleştirecek, diğerleri ise erişim izni bekleyecektir. Kısaca programınızın özeti;

- Kullanıcıdan islem\_sayisi, kuyruk\_uzunlugu verilerini al.

- Kullanıcıdan her işlem için ayrı ayrı isim al

- Bütün işlemleri start() fonksiyonu ile başlat,

- Programı bitir

İşlemler ise çalışırken şu yolu izler;

- mutex'i iste;

- izin verilince,

- eğer sayac değişkeni kuyruk\_uzunlugu kadarsa

- 1 saniye uyu,

- mutex'i bırak

- tekrar başa dön

- eğer sayac değişkeni kuyruk\_uzunlugu'ndan küçükse

- sayac değişkenini 1 artır

- mutex'i bırak

- işlemini gerçekleştirdiğini ekrana bastır

- mutex'i iste

- izin verilince

- sayac değişkenini 1 düşür

- mutex'i bırak

- bitir

Gördüğünüz gibi yukarıda "pseudo-code"u verilen ödev aslında işletim sisteminde işlemlerin nasıl çalışmaya alındığının basitleştirilmiş bir örneğidir.

Bu ödev için bizim size vermiş olduğumuz "process.java" isimli java dosyasının ilgili kısımlarını düzenleyip, "process.java" ismiyle sisteme yükleyeceksiniz. Temel kısımları içeren dosyayı biz sizin için hazırladık, sizin yapmanız gereken sadece verilen "comment"lerin olduğu kısımlara gerekli kodları yazmak olacaktır. (// \*\*\*\*\* ve // \*\*\*\* kısımları arasına gerekli kodları yazmanızı bekliyoruz)