



Öğrencinin;

ADI: Selin

SOYADI: Daldaban

NO: 1421221014

BÖLÜM: Bilgisayar Mühendisliği

Projenin;

KONUSU: Thread Senkronizasyonu

Dersin;

ADI: İşletim Sistemleri

EĞİTMEN:

Prof. Dr. A. Yılmaz ÇAMURCU

Ar. Gör. Samet KAYA

Ar. Gör. Okan KARA

İçindekiler

1. Özet	3
2. Proje Konusu	4
3. Proje Çıktıları ve Başarı Ölçütleri	4
4. Proje Süresince Yapılanlar	5
5. Kaynakça	5

1 Özet

Thread nedir?

Bilgisayar bilimlerinde özellikle işletim sistemi (operating systems) konusunda kullanılan önemli terimlerden birisidir. Bir işletim sisteminde ya da yazılan bir programda birden fazla işin aynı anda yapılması için kullanılırlar. [1]

Named Pipe Nedir?

Pipe, Unix sistemlerinde dosya sistemi ile bütünleştirilmiştir. Pipe yönteminde temel olarak bir process pipe'a yazma yaparken bir process de okuma yapar. Okuma işlemi FIFO sistemine göre yapılır. Pipe kullanımında senkronizasyon bloke yöntemiyle sağlanabilir. Okuyan process pipe boşsa bloke olur. Pipe'ın bir uzunluğu vardır. Yazan taraf pipe doluysa pipe'ta yer boşalana kadar bloke olur. Pipe'tan okuma yapıldığında yer açılır. İşlem yazan tarafın pipe'ı kapatmasıyla sonlandırılır. Bu işlemi okuyan taraf sanki dosyanın sonuna gelmiş de 0 byte okunmuş gibi algılar. Pipe'a okuma ve yazma yapmak için yine **read** ve **write** fonksiyonları kullanılır. [3]

İşlem sıraları büyük önem taşır. Dikkat edilmediği takdirde yazma ve okuma işlemleri karışır.

Mutex Nedir?

Mutex neredeyse her türlü işletim sisteminde olan bir senkronizasyon nesnesidir. Mutex bir thread tarafından ele geçirilir. Mutex'i ele geçiren thread mutex'i istediği zaman serbest bırakabilmektedir. Genellikle sistemlerde bir thread mutex'i ele geçirdiği zaman başka bir thread bu mutex'i ele geçirmeye çalışınca bloke olur. Mutex'in sahibi olan thread mutex nesnesini bırakana kadar thread bekler. Bazı sistemlerde (örneğin Win32 sistemlerinde) mutex nesnesi aynı thread tarafından birden fazla kez ele geçirilmektedir. Bu durumda thread'in mutex'i bırakması için yine ele geçirme sayısı kadar bırakma işlemi yapılması gerekir.

Mutex nesnesinin yaratılması için **pthread_mutex_t** türünden bir nesne tanımlanır.

int pthread_mutex_init(pthread_mutex_t *mutex, pthread_mutex_attr_t *attr);

Fonksiyonun birinci parametresi mutex nesnesinin adresi, ikinci parametresi mutex özelliklerinin değerlerine ilişkin yapının adresidir.

Bir thread mutex nesnesini **pthread_mutex_lock** fonksiyonu ile ele geçirir. Thread mutex nesnesinin **pthread_mutex_unlock** ile bırakmaktadır. Mutex kullanılarak tipik bir kritik kod şöyle oluşturulur:

pthread_mutex_lock(...); Kilitlemek için kullanılır.

pthread_mutex_unlock(...); Kilidi açmak için kullanılır.

int pthread_mutex_lock(pthread_mutex_t *mutex);

int pthread_mutex_unlock(pthread_mutex_t *mutex); [2]

2 Proje Konusu

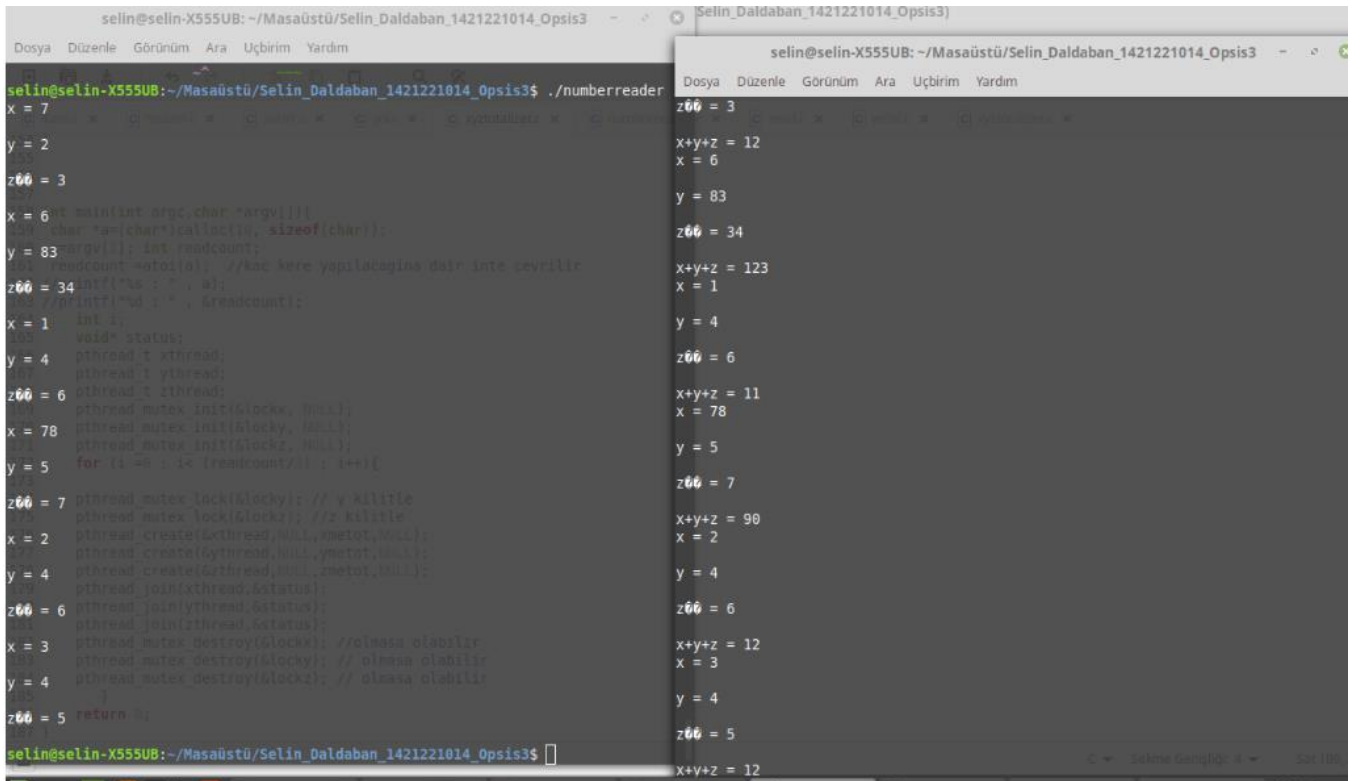
Xyztotalizer ve numberreader isimli iki adet c dosyası ile thread namedpipe, loc işlemleri kullanılarak istenenler gerçekleştirilmelidir.

X, y ve z isimli 3 adet thread olmalıdır. Numberreader isimli c dosyası numbers.txt dosyasından numaraları okuyup, xyztotalizer dan gelen sırasıyla x, y, z threadlerine okuma yapıp değerleri xyztotalizera yazıp, xyztotalizerda da x y z yi toplamalı ve her seferinde toplamı sıfırlamalıdır. Thread, namedpipe ve loc işlemleri doğru bir şekilde kullanılmalıdır.

3 Proje Çıktıları ve Başarı Ölçütleri

X, Y, Z threadleri ile değerleri alınıp toplama işlemleri doğru şekilde yapılmaktadır.

Eksik olan yer; satır numaralarını numberreadera gönderemedim.



```
selin@selin-X555UB: ~/Masaüstü/Selin_Daldaban_1421221014_Opsis3$ ./numberreader
x = 7
y = 2
z00 = 3
x = 6
y = 83
z00 = 34
x = 1
y = 4
z00 = 6
x = 78
y = 5
z00 = 7
x = 2
y = 4
z00 = 6
x = 3
y = 4
z00 = 5
selin@selin-X555UB: ~/Masaüstü/Selin_Daldaban_1421221014_Opsis3$
```

```
selin@selin-X555UB: ~/Masaüstü/Selin_Daldaban_1421221014_Opsis3$ ./xyztotalizer
z00 = 3
x+y+z = 12
x = 6
y = 83
z00 = 34
x+y+z = 123
x = 1
y = 4
z00 = 6
x+y+z = 11
x = 78
y = 5
z00 = 7
x+y+z = 90
x = 2
y = 4
z00 = 6
x+y+z = 12
x = 3
y = 4
z00 = 5
x+y+z = 12
```

4 Proje Süresince Yapılanlar

Namedpipe ile iki c dosyası arasında veri geçişleri yaptım. X Y Z threadlerini lock ve unlock yapısı ile sırasıyla çalıştırıp readernumberdan gelen değerlerini aldım ve toplama işlemlerini doğru şekilde yaptım.

5 Kaynakça

- [1] <http://bilgisayarkavramlari.sadievrenseker.com/2010/03/22/thread-iplik/>
- [2] <http://www.bilgisayardershanesi.com/Y5305-mutex-kullanimi.html>
- [3] <http://www.bilgisayardershanesi.com/Y5286-pipe-kullanimi.html>