

MIDTERM PROJECT

Bu projede üç adet program yazmamız gerekiyor. Bu programlar; **timeServer**, **seeWhat** ve **showResult**.

Program 1 – timeServer

Bu program bizim ana server'ımız olacak.
Çalıştırılma şekli:

./timeServer <ticks in miliseconds> <n> <mainpipeName>

<ticks in miliseconds> : Serverın kaç milisaniyede bir clientdan gelen requestleri kontrol edeceği değer.

<n> : İnteger bir değer. Server her client için $2n \times 2n$ invertible matris oluşturacak.

<mainpipeName> : Server ile clientin haberleşeceği ana fifonun adı.

Bu programın yapması gerekenler.

-- Her client için bir proses oluşturacak ve o prosesler her client için $2n \times 2n$ invertible random bir matris oluşturacak ve clienta gönderecek. Client request sinyalini ana fifo üzerinden gönderecek ve server client için oluşturduğu matrisi başka bir fifo üzerinden gönderecek.

-- **Bu programı oluşturması gereken log dosyaları ve log dosyasına yazılması gereken bilgiler.**

Bu programın 1 adet log dosyası oluşturması gerekiyor. Log dosyasına

Matrisin oluşturulduğu zaman(milisaniya olarak)

Clientın pid'si

oluşturulan matrisin determinanı.

Program 2 – seeWhat

Bu program bizim clientımız olacak.Birden fazla çalıştırılabilir.

Çalıştırılma şekli:

./seeWhat <mainpipeName>

Bu programın yapması gerekenler

Bu program serverdan veri($2n \times 2n$ invertible matris) alabilmek için servera request sinyali gönderecek. Aynı zamanda ana fifo üzerinden pid sini de göndermesi gerekiyor.Serverdan matrisi alacak ve bu matrisle ilgili bazı işlemler yapacak.Bu işlemleri yapmak için en az 2 tane proses oluşturmali.

Oluşturacağı proseslerden biri $n \times n$ lik shifted inverse matrisi bulacak ve **result1** i hesaplayacak.

Result1 = det(orjinal matris) – det(shifted inverse matris)

Diğer proses 2d convolution matris oluşturak ve **result2** yi hesaplayacak.

Result2 = det(orjinsl matris) – det(2d conconvolution)

Program result1 ve result2 nin ne kadar zamanda(time elapsed) hesaplandığını bulaması gerekiyor.

Shifted Inverse matris

$n \times n$	$n \times n$
$n \times n$	$n \times n$

Orjinal matrisi n x n lik 4 matrise bölüp her bir bölümün tersini alarak bulunduğu yere yazıyoruz.Oluşan yeni matris bizim shifted inverse matrisimiz oluyor.

2d convolution Matris

Bu matrisin nasıl oluşturulduğunu öğren. Raporda 2d convolutionun nerde kullanıldığı yazılacak.

Convolutionda kullanılacak matrisler

Kernel matrix

Skew-symetric matrix1

Skew-symetric matrix2

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & -1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}$$

Bu programın oluşturması gereken log dosyaları ve içerikleri

Bu program her işlem için bir tane log dosyası oluşturacak.

Her log dosyasına işlemler sonucu oluşan matrisler matlab formatında yazılacak
(Orijinal matris, shifted inverse matris, 2d convolution matris)

Program 3 – showResult

Programın çalıştırılma şekli

./ShowResult (argümentsiz)

Bu program sonuçları yazacak programımız olacak

Her clientdan result1, result2 ve clientın pidsini alacak ve onları dosyasına ve ekrana aşağıdaki gibi yazdıracak.

Ekrana yazılacak bilgiler

pid	Result1	Result2
.	.	.
.	.	.
.	.	.

Log dosyasına yazılacak bilgiler

m1	pid of client
result1,	time elapsed
result2,	time elapsed
.	.
.	.
.	.

Main Requirements

Ctrl + c sinyalinin handle edilmesi gerekiyor.Üç programdan herhangi birinin çalıştığı terminalden Ctrl c ye basılırsa diğer programların da sonlanması gerekli ve sinyalin geldiği zamanı ve sonlandığını log dosyasına yazması gerekiyor.

Ctrl + c sinyalinin geldiği programın log dosyasına sinyalin geldiği bilgisi yazılacak, diğer programlara kill sinyalinin geldiği bilgisi yazılacak.

Ödev için rapor hazırlanacak ve rapora projeyi nasıl hazırladığımızı yazacağız. Ayrıca raporda projenin birkaç farklı örnekle çalıştırılmış sonuçları olacak.

Ödevin gönderilmesi gereken format

Öğrenci xx

```
seewhat
timerServer
showResult
log \
  log 1
  log 2
  .
  .
```