

Gebze Technical University  
Computer Engineering

CSE 222  
2017 Spring

HOMEWORK IV REPORT

STUDENT NAME  
LÜTFULLAH TÜRKER

STUDENT NUMBER  
141044050

Course Assistant:  
Nur Banu Albayrak

## **1. Test Cases**

## **2. Running and Results**

## 1. Test Cases

Senaryo	Test	Beklenen Sonuç	Gerçek Sonuç
Program test.csv isimli bir dosyayı okur.Ve bu dosyadan ',' lerle ayrılmış olan her girdiyi alır ve Stack e veya Queue ye doldurur.test.csv nin her satırında bir tipin girdileri vardır.(int,Double vb.)	test.csv de 4 satır ve 4 tip sıralı şekilde int,Double,char,String şeklinde olmalıdır.Farklı düzende test.csv dosyası verilirse ne olur ?	Stack veya Queue yi doğru şekilde dolduramaması veya exception fırlatılması beklenir.	Bazı durumlarda doğru şekilde çalışıp result csv yi doğru yazmakta Fakat Test kısmında belirtilen şekilde girilmesi hata olmaması için daha uygundur.

## 2. Running and Results

### 1.Soru için :

```
68         end = System.nanoTime() - start;
69         System.out.println("StackC running time is ==> "+end/pow(10,9)+" seconds.\n");
70
71         start = System.nanoTime();
72         fill.readAndFill(D,1);
73         writeResultCSV(D);
74     }
```

Output - DataStructures HW4.1 (run) x

```
run:
StackA running time is ==> 0.044062139 seconds.
StackB running time is ==> 0.010651161 seconds.
StackC running time is ==> 0.009959016 seconds.
StackD running time is ==> 0.006929596 seconds.
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
68
69
70     end = System.nanoTime() - start;
71     System.out.println("StackC running time is ==> "+end/pow(10,9)+" seconds.\n");
72
73     start = System.nanoTime();
74     fill.readAndFill(D,1);
75     writeResultCSV(D);
```

Output - DataStructures HW4.1 (run) x

```
run:
StackA running time is ==> 0.036049211 seconds.
StackB running time is ==> 0.018481359 seconds.
StackC running time is ==> 0.007640492 seconds.
StackD running time is ==> 0.009058215 seconds.
BUILD SUCCESSFUL (total time: 0 seconds)
```

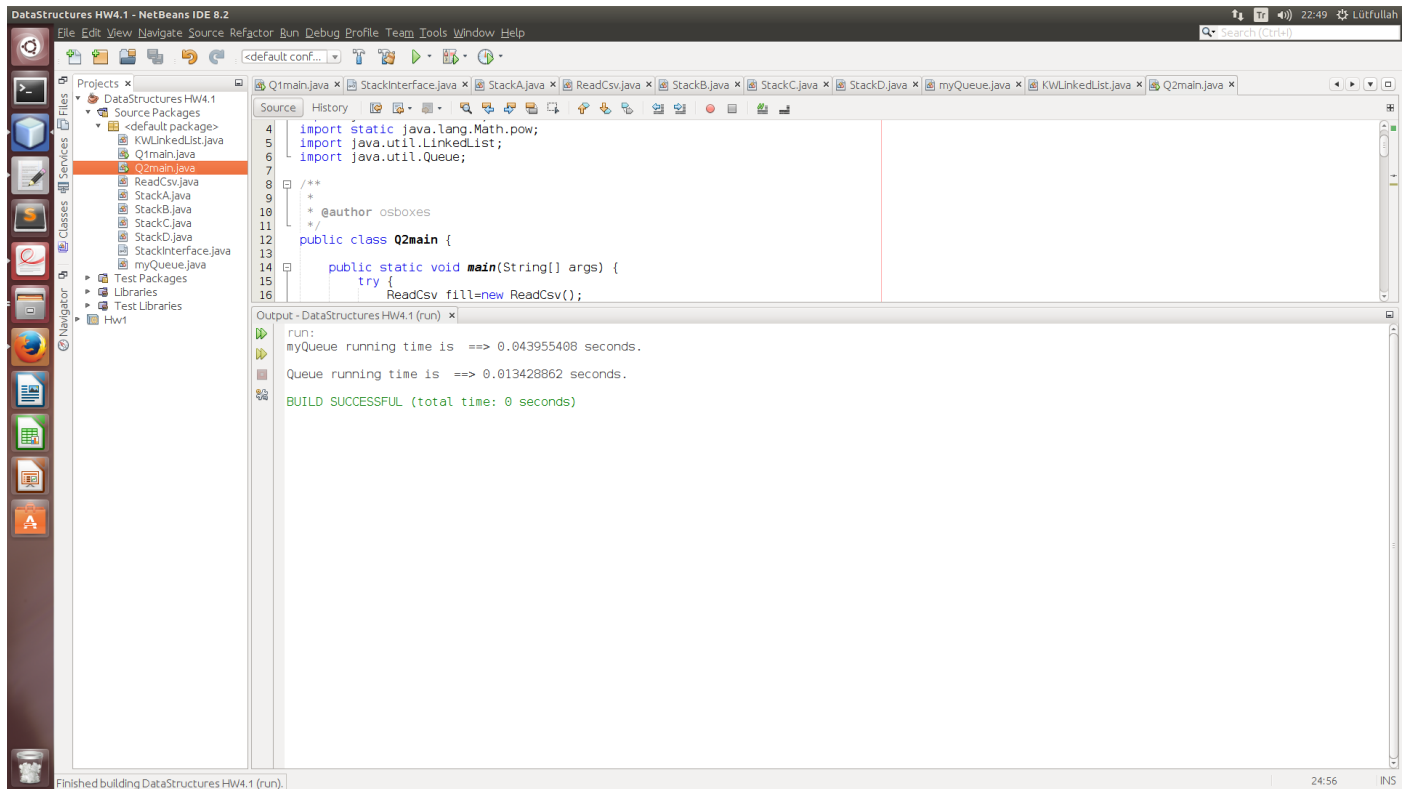
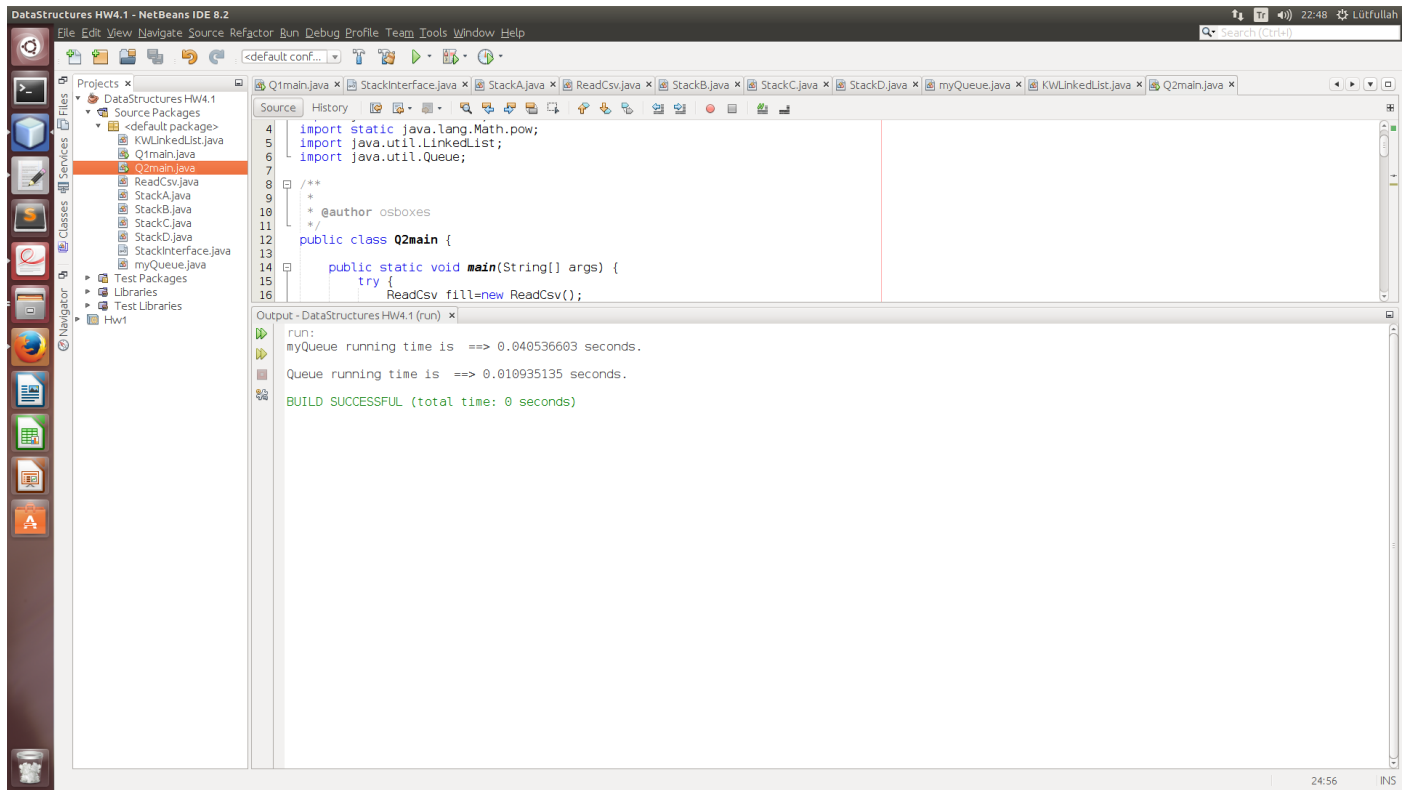
```
68
69
70     end = System.nanoTime() - start;
71     System.out.println("StackC running time is ==> "+end/pow(10,9)+" seconds.\n");
72
73     start = System.nanoTime();
74     fill.readAndFill(D,1);
75     writeResultCSV(D);
```

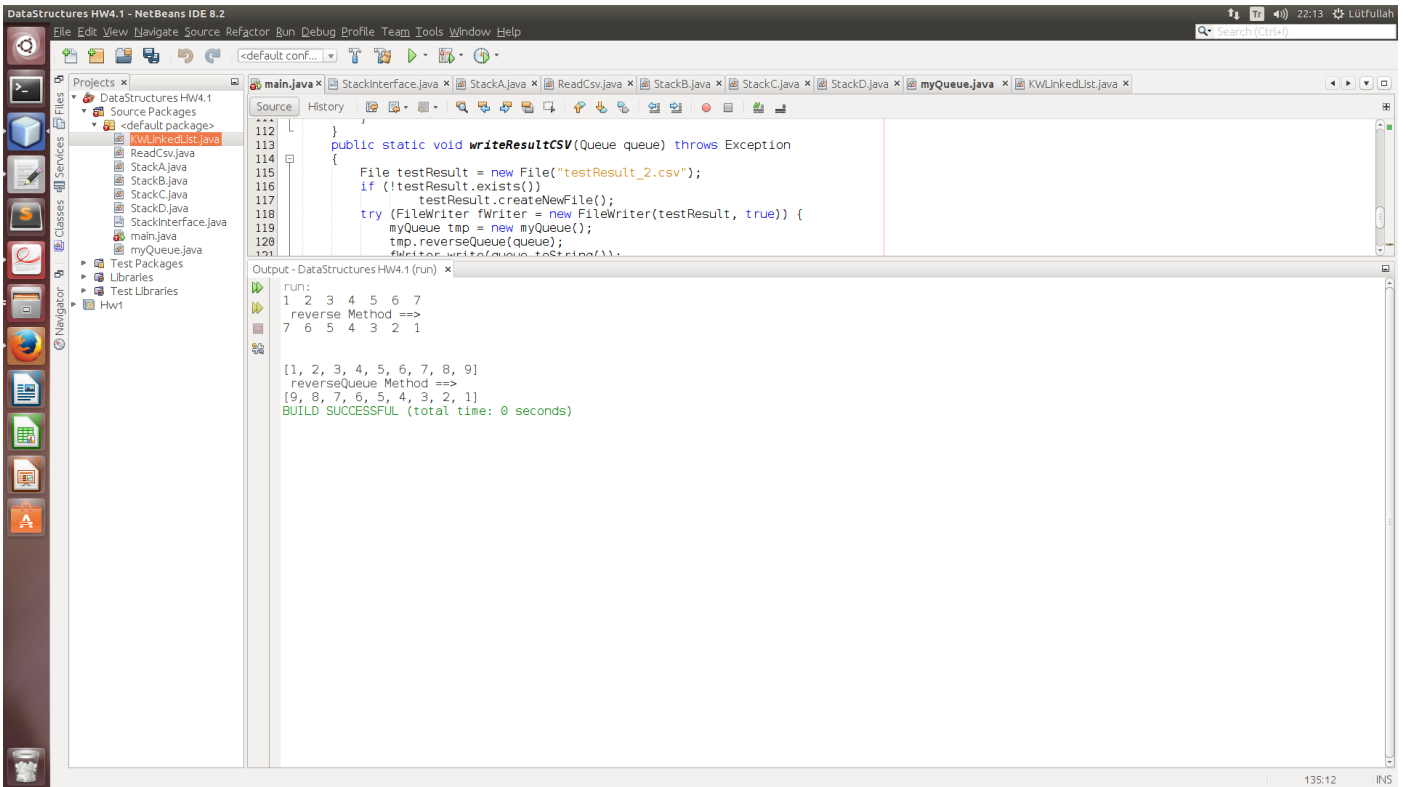
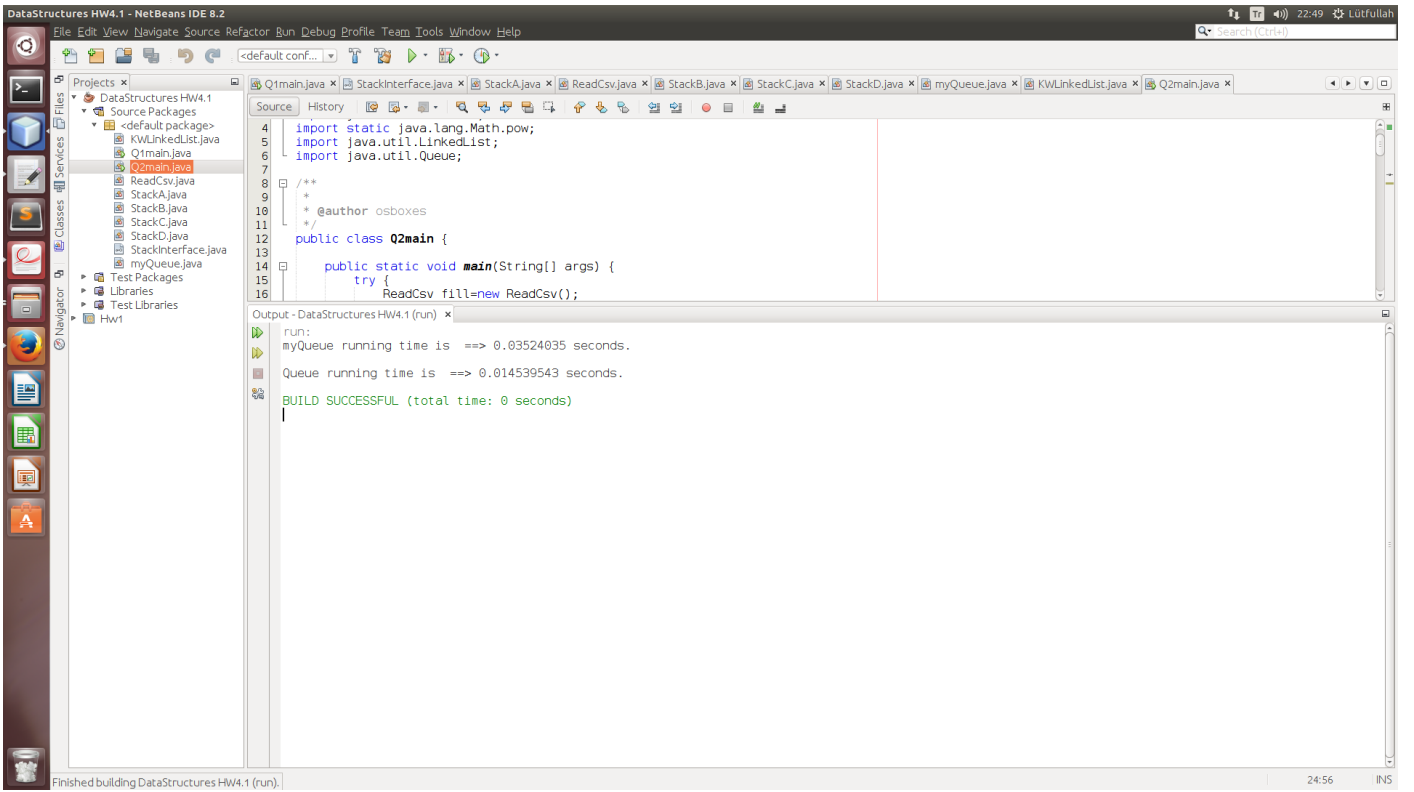
Output - DataStructures HW4.1 (run) x

```
run:
StackA running time is ==> 0.046758328 seconds.
StackB running time is ==> 0.012838992 seconds.
StackC running time is ==> 0.009895849 seconds.
StackD running time is ==> 0.007438772 seconds.
BUILD SUCCESSFUL (total time: 0 seconds)
```

Sonuçlarda görüldüğü gibi en hızlı StackD yani Queue kullanılan Stack ve en yavaş da belirgin bir farkla ArrayList i extend ettiğimiz StackA oldu.2 Array kullanan Stack de en yavaş olması normal bir durum.Node yapısı kullanan classlar bu işlemlerde daha hızlı.

## 2.Soru için :





Yukarıda görüldüğü gibi Queue class ı myQueue class ımızdan daha hızlı çalıştı.(Normal olarak).