Hazal Gonen 131044028

Bu program verilen input txtsinde olan ifadeleri okur ve o ifadeleri assembly diline cevirir.

SISTEM GEREKSINIMLERI:

Ilk olarak bir input dosyasi olusturulmalıdır.

Daha sonra Expressions objesi olusturulmali ve bu objeye input

dosyasının adı gonderilmelidir.

Expression objesi olusturulduktan sonra execute metodu çağırılmalı.

Daha sonra Expressions objesinin convertInfixToPostfix metodu cagirilarak ifade postfix hale cevrilir

En son Expressions objesinin convertPostfixToAsm metodu cagirilarak postfix ifade assemblye cevrilir ve output.asm dosyasına kaydedilir.

PROBLEM COZUM YONTEMLERI:

Program basladığında ilk olarak Expressions objesi olusuyor.

Bu objenin constructoruna input dosyasının pathi gonderiliyor

Constructorda read yapılarak context isimli string arrayine(MyArrayList) eklenir.

Her satır arrayin 1 elemanını temsil eder.

Daha sonra mainde çağırılan execute metodu ile bu stringler Expression tipinde ki arrayliste doldurulur

Bu doldurma yapılırken her Expression tipine uygun olarak doldurulur

Orneğin a bir Variable 6 bir temporaryVariable + bir operatordur.

Yani son olarak Expression tiplerinden oluşan bir Expression arraylistim oluşur ve bu arraylist infix'dir.

Sıra bu infixi postfixe çevirecek olan metodu çağırmaya geldi

Mainde Expressions objesi convertInfixtoPostfix metodunu çağırır.

Bu metod ise içinde InfixToPostfix objesini olusturup her satır için convert metodunu çağırır.

ve dönen Expressionı postfixExpression arrayListine ekler.

postfix hale çevrilirken önceliklere dikkat edilir.

postfix hale çevrilirken convert metodu içinde bir operator stacki tutar ve operator geldiği zaman bu stack push yapılır ancak öncelik sırası önemli eğer önceliği fazla olan bir operator gelirse postfixe ekleme yapilir.

Boylece a = 4 infixi a 4 = haline doner.

Sıra geldi bunu assemblye çevirmeye

Mainde convertPostfixtoAsm metodu çağırılır

Bu metod içinde PostfixToAsm objesi olusturur ve her satiri tek tek convert metoduna gonderir.

Convert metodu ise içinde operand stack tutar

Her ifadeye tek tek bakar ve eğer operand ise stack doldurulur operator ise boşaltılır işlem yapilip işlemin sonucu tekrar doldurlur

Ve bu sırada assembly kodları dosyaya ve ekrana yazılır

MAINDE YAZILAN TEST SENARYOLARI:

Mainde direk input dosyasinin pathi verilerek Expression objesi oluşturulur.

Bu objenin execute metodu çağırılır

daha sonra convertInfixToPostfix metodu çağırılır

daha sonra da convertPostfixToAsm metodu çağırılır

Senaryo inputun içeriğine göre değişir

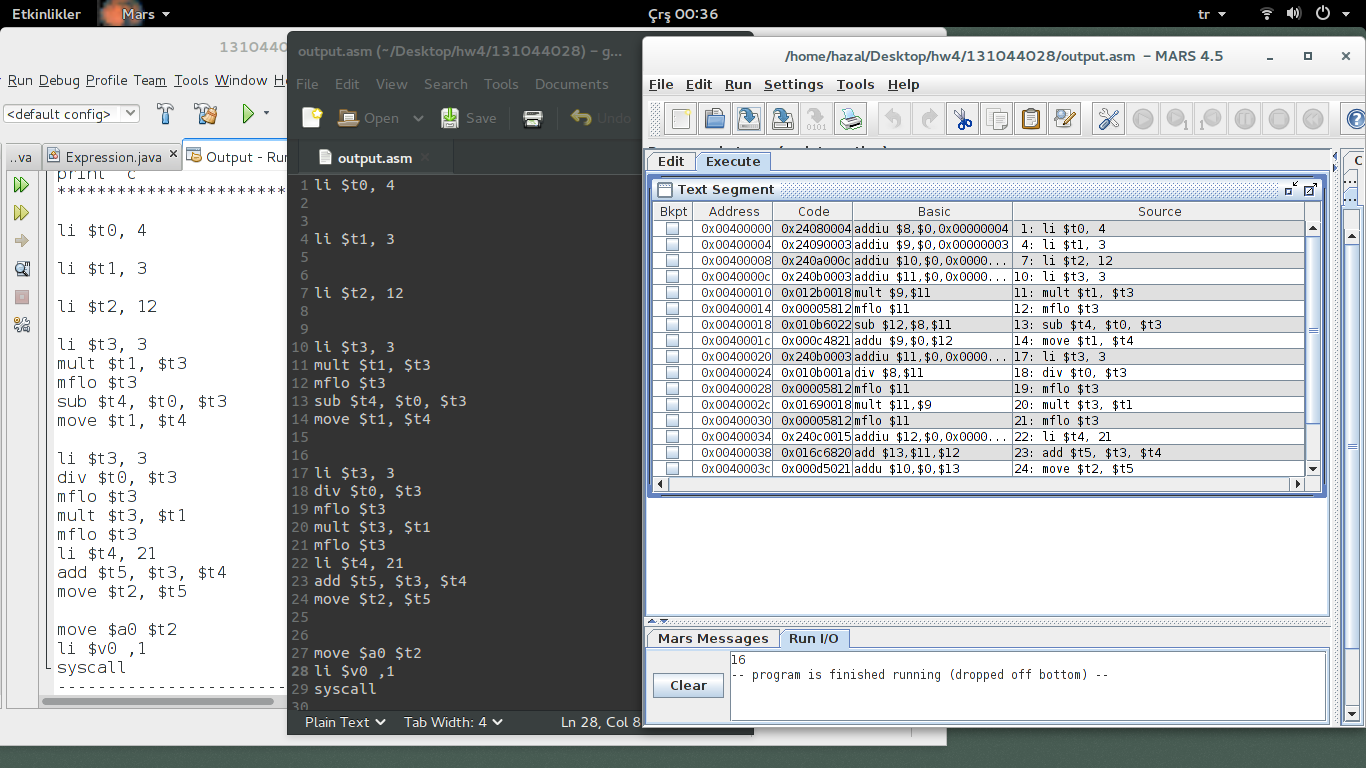
Program çalıştıktan sonra

İlk ekran ekrana basılan output

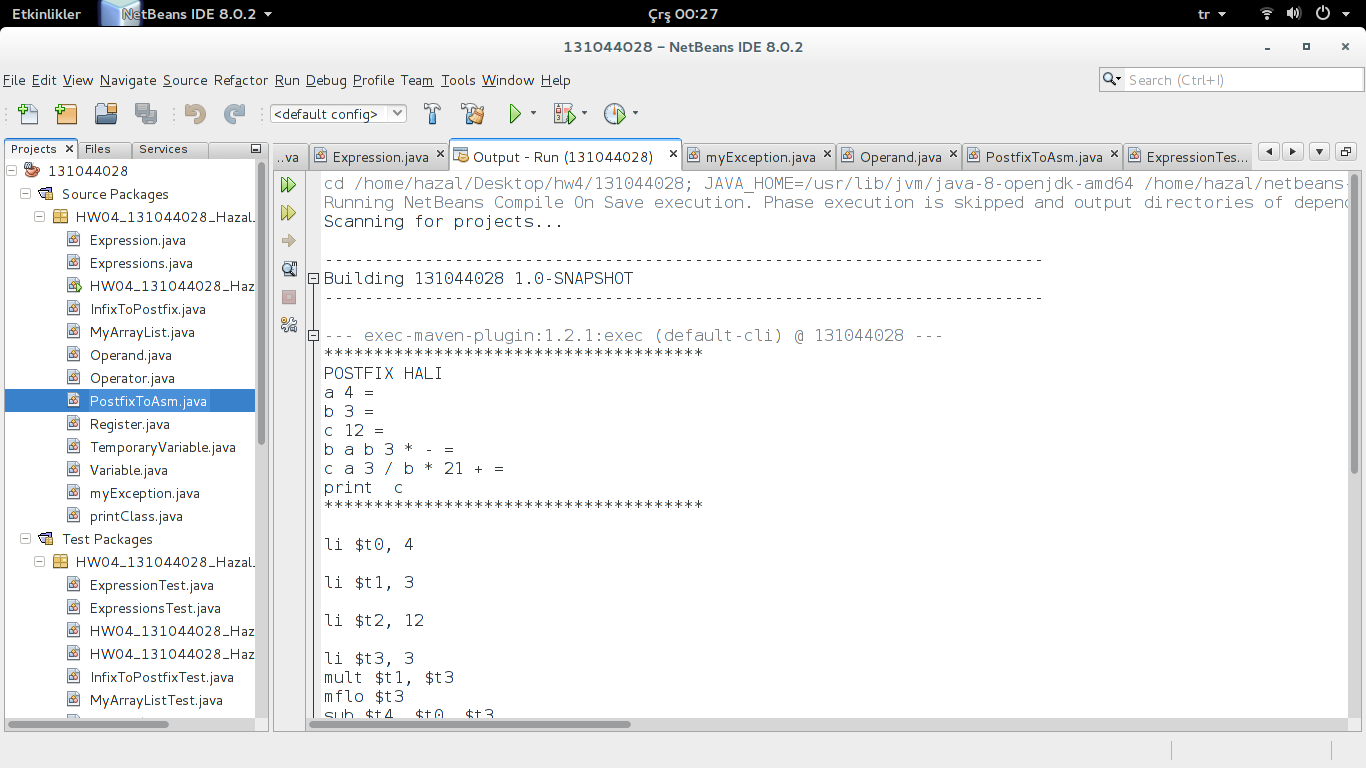
2. ekran output.asm dosyası

3.ekran mars editoru

ve cevap 16 çıkmalıydı.



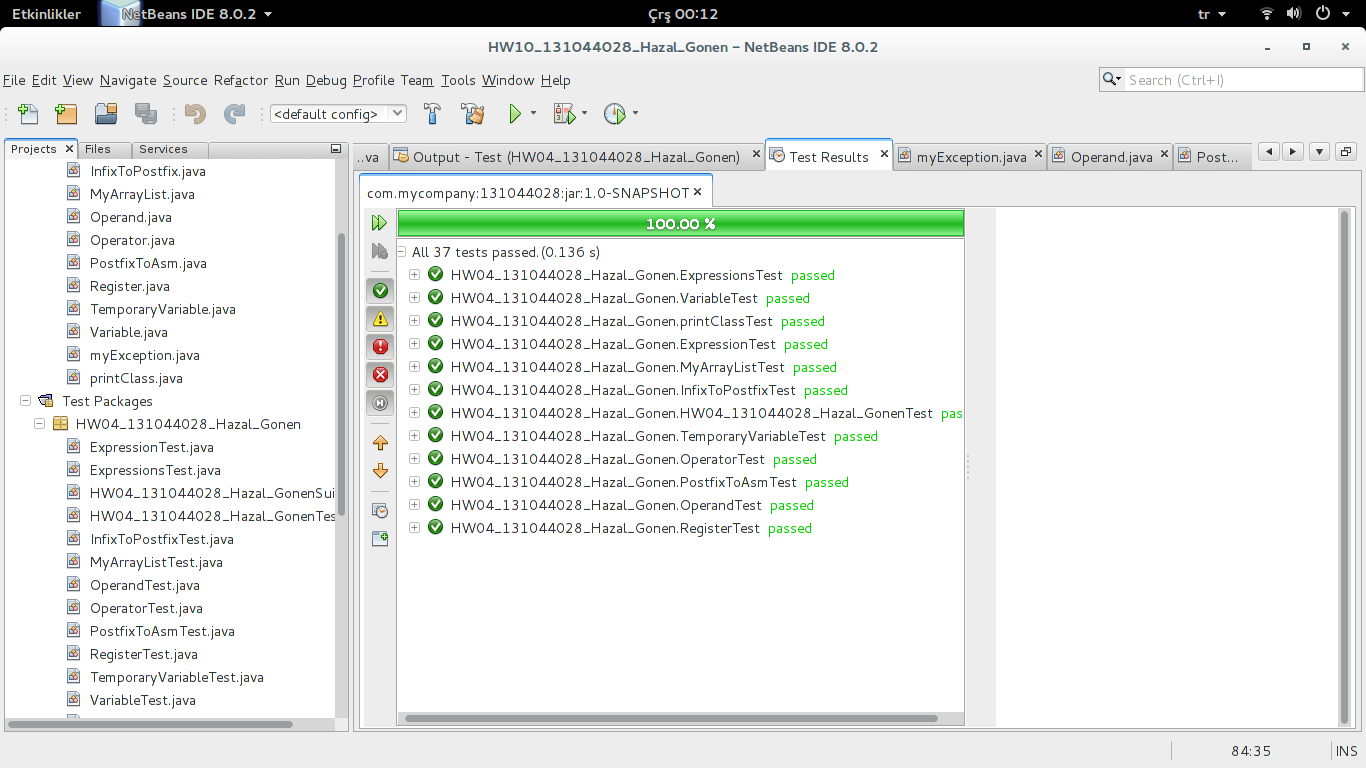
inputun postfix halini ekrana bastırdım



UML DIYAGRAMLARI(Okunmuyor pdf olarak ekledim)

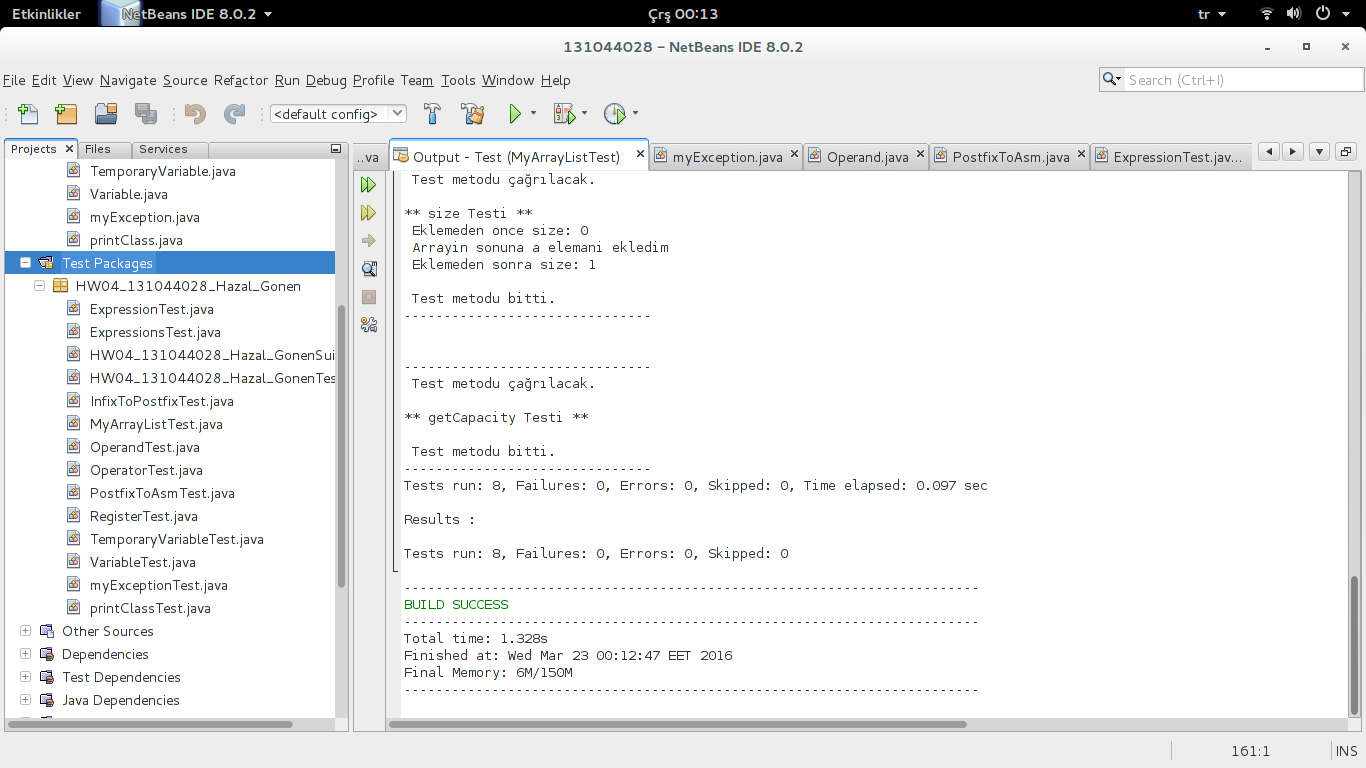
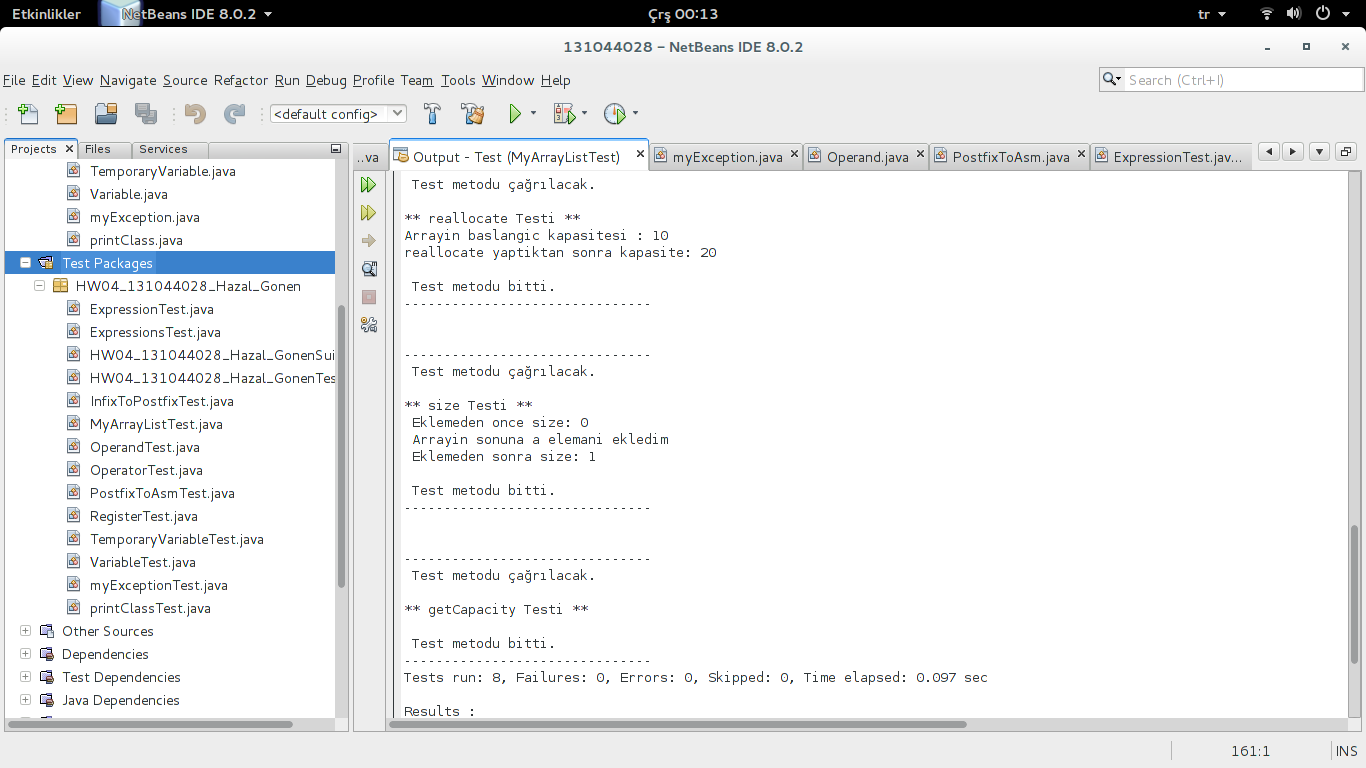
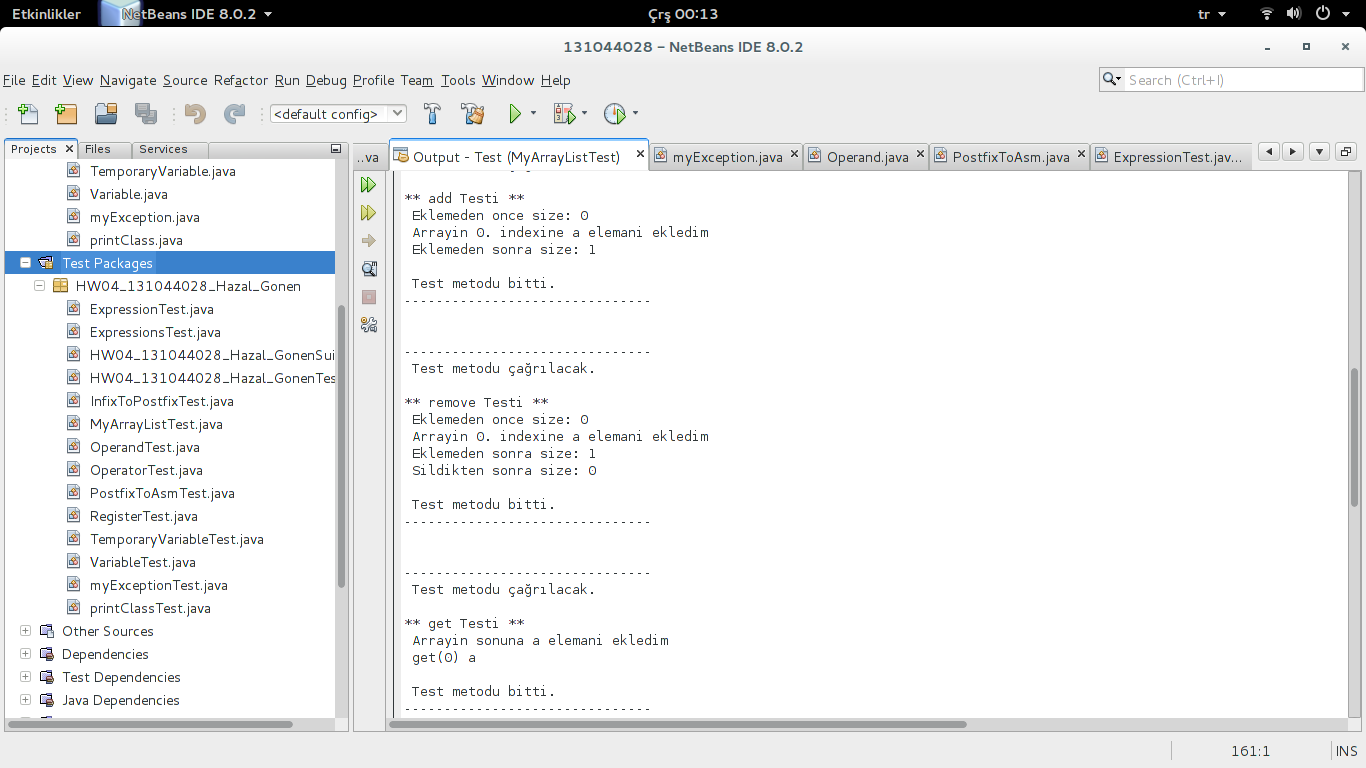
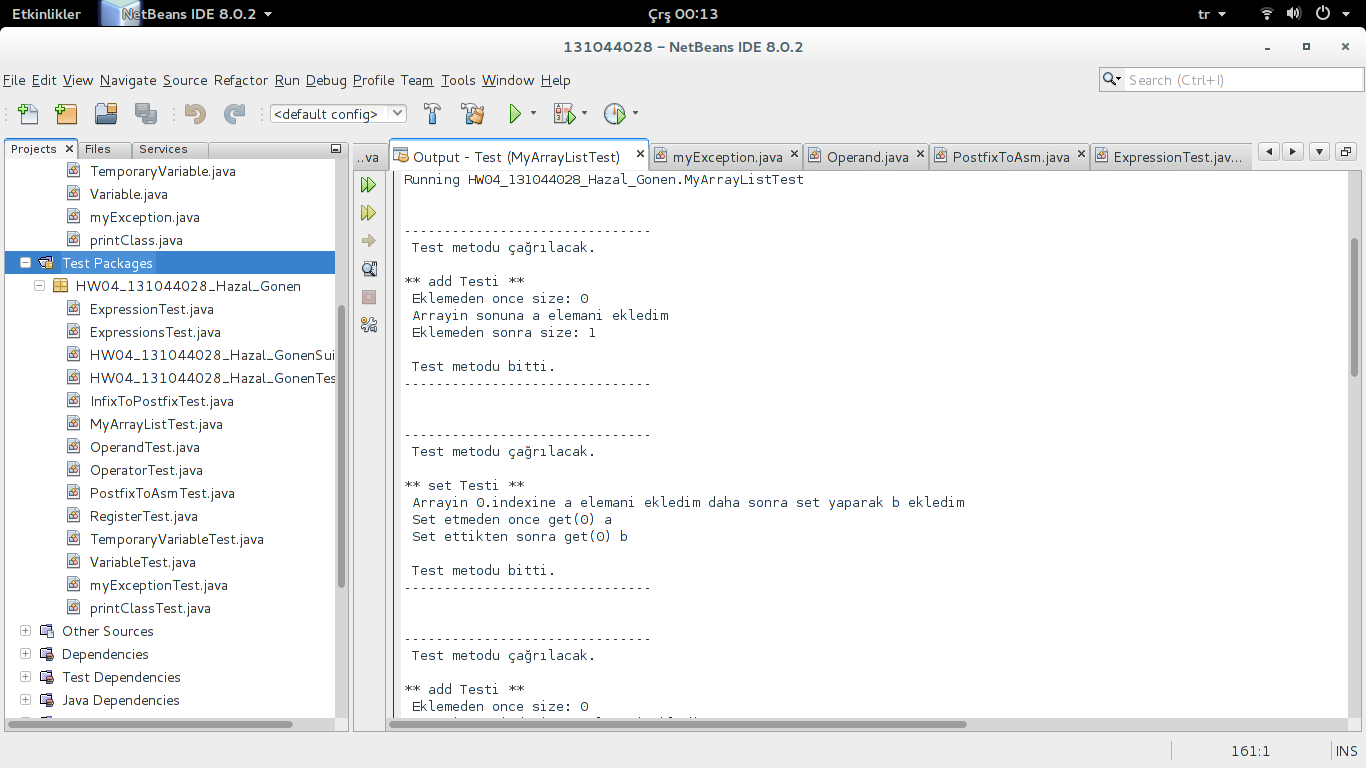
JUNIT TEST SCREENSHOTLARI

Butun testler başarılı



Bazı testlerin screen shotları var hepsini koymadım

MyArrayList test



Expressions Testi

