

**Derleyici Tasarım**

**Ali Hamitoğlu**

**Hazırlayanlar:**

Yazılım Mühendisliği – 4. Sınıf

Burak Kırbıyık

Abdullah Halid Uslu

İbrahim Özsoy

**Kullandığımız Grammer**

P → {C} '.'

C → I | W | A | Ç | G

I → '[' E '?' C{ C } ':' C{ C } ']' | '[' E '?' C{C} ']'

W → '{' E '?' C{C} '}'

A → K '=' E ';'

Ç → '<' E ';'

G → '>' K ';'

E → T {('+' | '-') T}

T → U {('\*' | '/' | '%') U}

U → F '^' U | F

F → '(' E ')' | K | R

K → 'a' | 'b' | … | 'z'

R → '0' | '1' | … | '9'

Programımızı adım adım inceleyecek olursak ilk olarak programımız Java dilinde yazılmıştır. Main metodumuz ile işlem yaptıracağımız fonksiyonu çağırıyoruz (Programı başlat). Sonra fonksiyonun bulunduğu class’tan başlattığımız (çağırdığımız) P() fonksiyonunda bir getToken fonksiyonu çağırıyoruz. Bu fonksiyon kullanıcıdan değer istemek için yazılmış fonksiyon. Bu fonksiyon bize bir String değer döndürüyor. P() fonksiyonuna geçmeden önce global de tanımlanan bir değişkenimiz daha var(abc) bu değişken while döngüsünde olup olmadığımızı bize belirtiyor. Şimdi fonksiyonlara göz atalım.

P() fonksiyonu, bu fonksiyon programın başlayıp başlamadığını bizeler gösterir ve getToken() fonksiyonunu çağırarak kullanıcıdan bir değer ister. Girilen değeri char (yani karaktere) çevirerek if ile sorgularız. Sonuçlar;

‘.’ ise programı bitirir.

‘}’ ise programı fonksiyonunun kapama sembolünün geldiğini belirtir.

‘{’ ise C() fonksiyonunun çalışacağını belirtir.

C() fonksiyonu, bu fonksiyon programda hangi işlemi yapacağımı anladığımız bir fonksiyon. Yine P() fonksiyonundaki gibi önce bir değer istiyoruz kullanıcıdan ardında girilen değeri sorguluyoruz. Sonuçlar;

‘.’ ise programı bitirir.

‘[’ ise I() fonksiyonun çağırır. Fakat çağırmadan önce koşul için bir int değer alır. Sonra I() fonksiyonuna gider.

‘{’ ise W() fonksiyonun çağırır. Fakat çağırmadan önce while’a girdiğini belirtmek için abc değişkenini true yapıyoruz ve koşul için bir int değer alır. Sonra W() fonksiyonuna gider.

‘<’ ise O() fonksiyonun çağırır.

‘>’ ise G() fonksiyonun çağırır.

‘=’ ise A() fonksiyonun çağırır.

‘]’ ise I() fonksiyonun bittiğini belirtir ve C() fonksiyonu geri çağırır.

‘}’ ise W() fonksiyonun bittiğini belirtir ve C() fonksiyonu geri çağırır.

‘}’ ise C() fonksiyonun bittiğini belirtir ve P() fonksiyonu geri çağırır.

I() fonksiyonu, bu fonksiyon bir if koşuluna girdiğini belirtir ve bir karakter istenir. Girilen karakter ‘?’ ise koşulun doğruluğuna bakıyorız bunun için ise parametre olarak gönderilen int değeri 0’a eşit mi değil mi kontrol ettik. Duruma göre ya if in içi ya da else kısmı çalışır. Yeni bir Cümle girişi ve ya çıkışı sorgulanır ve ya if ten çıktığına dahil bir değer olduğunu gösterir.

W() fonksiyonu, bu fonksiyon bir while koşuluna girdiğini belirtir ve bir karakter istenir. Girilen karakter ‘?’ ise koşulun doğruluğuna bakıyorız bunun için ise parametre olarak gönderilen int değeri 0’a eşit mi değil mi kontrol ettik. Duruma göre ya While in içi ya da else kısmı çalışır. Yeni bir Cümle girişi ve ya çıkışı sorgulanır ve ya while den çıktığına dahil bir değer olduğunu gösterir.

O() fonksiyonu, bu fonksiyon bir çıkış fonksiyonuna girdiğini belirtir ve girilen değer ekrana yazdırılır.

G() fonksiyonu, bu fonksiyon bir çıkış fonksiyonuna girdiğini belirtir ve girilen değer ekrana yazdırılır.

A() fonksiyonu, bu fonksiyon bir atama fonksiyonuna girdiğini belirtir.

E() fonksiyonu, bu fonksiyon bir toplama, çıkarma yapan fonksiyondur. 2 int, 1 operatör için karakter alıyoruz. ve işlemi gerçekleştirip return ediyoruz.

T() fonksiyonu, bu fonksiyon bir çarpma bölme fonksiyonudur. 1 int ve 1 operatör için karakter alıyoruz. E’den gelen sonucu T de aldığımız sonuç ile işlemini gerçekleştirip sonucu return ediyoruz.

U() fonksiyonu, bu fonksiyon bir üst alma fonksiyonudur. T’den gelen sonucu karesini alıp sonucu return ediyoruz.

<https://paste.ubuntu.com/p/Pf4NdMVY3P/>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**KOD KISIMI**

package derleyiciTasarim;

import java.io.ObjectInputStream.GetField;

import java.util.Scanner;

import javax.xml.bind.ParseConversionEvent;

public class Derleyici {

public static void main(String[] args) {

/\*

Burası bizim programızmız baştan kısmı burada

token isimli class a bağlanıp ilk olarak P()

fonksiyonunu çağırıyoruz.

\*/

Token token = new Token();

token.P();

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**package** derleyiciTasarim;

**import** java.util.Scanner;

**public** **class** Token {

/\*

Her adımda kullanıcıdan bir getToken ile bir karakter alıyoruz ve

girilen her fonksiyonda gelen string'i char'a çevirip ilk karakyterini

alıp sorguluyoruz.

\*/

**public** **void** P() {

/\*

P() fonksiyonumuzda Programın bitirme '}' sembolünü ve programın

'.' durdurma sembolünü girilmesi beklenir ve ona göre bir işlem

yaptırır. Program fonksiyonunun dışında '{' sembolü ile cümle

fonksiyonu çağırırız.

\*/

String str;

**char** ch;

str = getToken();

ch = str.charAt(0);

**if**(ch == '.') {

System.***out***.println("Program Bitti");

System.*exit*(0);

}**else** **if**(ch == '{') {

System.***out***.println("Cümle fonksiyonu");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Program fonksiyonu bitti");

P();

}**else** {

System.***out***.println("Boşta kaldı iyimi :)");

}

}

Scanner sc = **new** Scanner(System.***in***);

**private** String getToken() {

/\*

getToken() fonksiyonumuz ile kullanıcıdan bir string alıyoruz

girilmesi beklenen tek bir karakterdir. Girilen string'i

return olarak döndürüyoruz.

\*/

String token;

System.***out***.println("Enter value :");

token = sc.next();

**return** token;

}

**private** **void** C() {

/\*

C() fonksiyonumuz ile kullanıcıdan hangi cümleye gitmek istediği

belirleyip o fonksiyonu çağırıyoruz. Girilmesi için beklenen

semboller '[','{','<','>','=' fonksiyonlara giriş içindir.

']' ve '}' semboller ile cümle veye if fonksiyonların bittiğini

gösterir.

\*/

String str;

**char** ch;

str = getToken();

ch = str.charAt(0);

**if**(ch == '.') {

System.***out***.println("Program Bitti");

System.*exit*(0);

}**else** **if**(ch == '[') {

System.***out***.println("If fonksiyonu");

**int** num,val;

System.***out***.println("If için bir sayı giriniz: ");

num = sc.nextInt();

val = E1(num);

I(val);

}**else** **if**(ch == '{') {

System.***out***.println("While fonksiyonu");

**int** num,val;

System.***out***.println("While için bir sayı giriniz: ");

num = sc.nextInt();

W(num);

}**else** **if**(ch == '<') {

System.***out***.println("Çıkış fonksiyonu");

O();

}**else** **if**(ch == '>') {

System.***out***.println("Giriş fonksiyonu");

G();

}**else** **if**(ch == '=') {

System.***out***.println("Atama fonksiyonu");

System.***out***.println("Atanacak bir String giriniz: ");

String t = sc.next();

A(t);

}**else** **if**(ch == ']') {

System.***out***.println("If fonksiyonu bitti");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

P();

}

}

**private** **void** I(**int** val) {

/\*

I() fonksiyonumuz ile gelen değişkeni koşula sokuyoruz ve koşul

yanlış ise else koşul doğru ise if in içindeki cümleler çalışıyor

ve sonunda işlem duruyor

\*/

String str;

**char** ch;

**int** num = val;

str = getToken();

ch = str.charAt(0);

**if**(ch == '?') {

**if**(num == 0) {

System.***out***.println("Koşul yanlış");

}**else** {

System.***out***.println("Koşul doğru");

}

System.***out***.println("If fonksiyonu koşul bitti");

I(num);

}**else** **if**(ch == ':') {

System.***out***.println("If fonksiyonu else mevcut");

I(num);

}**else** **if**(ch == ']') {

System.***out***.println("If fonksiyonu bitti");

C();

}**else** **if**(ch == '{') {

System.***out***.println("If fonksiyonundan cümleye geçiş");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

C();

}**else** **if**(ch == ']') {

System.***out***.println("If fonksiyonu bitti");

C();

}**else** {

System.***out***.println("Boşta kaldı iyimi :)");

}

}

**private** **void** W(**int** val) {

/\*

W() fonksiyonumuz ile gelen değişkeni koşula sokuyoruz ve koşul

yanlış olana kadar while dönüyor ve sonunda işlem duruyor

\*/

String str;

**char** ch;

**int** x,num = val;

str = getToken();

ch = str.charAt(0);

**if**(ch == '}') {

System.***out***.println("While fonksiyonu bitti");

C();

}**else** **if**(ch == '?') {

**while**(num<5) {

x = E2(num);

**if**(x == 0) {

System.***out***.println("Koşul yanlış");

}**else** {

System.***out***.println("Koşul doğru");

System.***out***.println(num);

num++;

}

}

System.***out***.println("Koşul yanlış");

System.***out***.println("While fonksiyonu koşul bitti");

C();

}**else** **if**(ch == '{') {

System.***out***.println("While fonksiyonu cümle");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

C();

}**else** **if**(ch == ']') {

System.***out***.println("If fonksiyonu bitti");

C();

}**else** {

System.***out***.println("Boşta kaldı iyimi :)");

}

}

**private** **void** O() {

/\*

O() fonksiyonumuz ile daha önceden belirlediğim bir mesajı ekrana

yazdırıp fonksiyonun bitiriyoruz.

\*/

String str;

**char** ch;

str = getToken();

ch = str.charAt(0);

**if**(ch == ';') {

System.***out***.println("Program güzel çalışıyor :)");

System.***out***.println("Çıktı fonksiyonu bitti");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

C();

}**else** {

System.***out***.println("Çıktı fonksiyonu");

O();

}

}

**private** **void** G() {

/\*

G() fonksiyonumuz ile sırasıyle E(),T() ve U() fonksiyonlarını

çağırıyoruz ve çıktıları ekrana yazdırıp fonksiyonu bitiyoruz

\*/

String str;

**char** ch,o1,o2;

**int** x,y,z,esonuc,tsonuc,usonuc;

System.***out***.println("E() için iki int ve bir operatör giriniz: ");

x = sc.nextInt();

y = sc.nextInt();

str = sc.next();

o1 = str.charAt(0);

esonuc = E(x,y,o1);

System.***out***.println("T() için bir int ve bir operatör giriniz: ");

z = sc.nextInt();

str = sc.next();

o2 = str.charAt(0);

tsonuc = T(esonuc,z,o2);

usonuc = U(tsonuc);

str = getToken();

ch = str.charAt(0);

**if**(ch == ';') {

System.***out***.println("E() Sonuc: " + esonuc);

System.***out***.println("T() Sonuc: " + tsonuc);

System.***out***.println("Sonucun karesi: " + usonuc);

System.***out***.println("Girdi fonksiyonu bitti");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

C();

}**else** {

System.***out***.println("Girdi fonksiyonu");

G();

}

}

**private** **void** A(String text) {

/\*

A() fonksiyonumuz ile alınan değeri bir değişkene atıyoruz ve girilen

karakter ile atama yapılan değişkeni ekrana yazdırıp fonksiyonun

bitiriyoruz.

\*/

String str,d;

**char** ch;

d = text;

str = getToken();

ch = str.charAt(0);

**if**(ch == ';') {

System.***out***.println("d = " + d);

System.***out***.println("Atama fonksiyonu bitti");

C();

}**else** **if**(ch == '}') {

System.***out***.println("Cümle fonksiyonu bitti");

C();

}**else** {

System.***out***.println("Atama fonksiyonu değişken");

A(d);

}

}

**private** **int** E1(**int** num) {

/\*

E1() fonksiyonumuz ile alınan değeri sabit bir işleme sokarak

çıkan sonucu return ediyoruz

\*/

**int** a;

a = num - (7 - 2);

**return** a;

}

**private** **int** E2(**int** num) {

/\*

E2() fonksiyonumuz ile alınan değeri sabit bir işleme sokarak

çıkan sonucu return ediyoruz

\*/

**int** a;

a = (num \* 7) - 35;

**return** a;

}

**private** **int** E(**int** x,**int** y,**char** o) {

/\*

E() fonksiyonumuz ile alınan 2 değeri ve verilen operatörü ile

girilen operatörün işlemini yaptırıyoruz sonucu return ediyoruz.

\*/

**int** sonuc;

**if**(o == '+') {

sonuc = x + y;

}**else** {

sonuc = x - y;

}

**return** sonuc;

}

**private** **int** T(**int** x,**int** y,**char** o) {

/\*

T() fonksiyonumuz ile alınan 2 değeri ve verilen operatörü ile

girilen operatörün işlemini yaptırıyoruz sonucu return ediyoruz.

\*/

**int** sonuc;

**if**(o == '\*') {

sonuc = x \* y;

}**else** {

sonuc = x / y;

}

**return** sonuc;

}

**public** **int** U(**int** x) {

/\*

U() fonksiyonumuz ile alınan değerin karesini hesaplama

işlemi yapıp sonucu return ediyoruz.

\*/

**int** sonuc = 1;

**for**(**int** i=0;i<2;i++) {

sonuc \*= x;

}

**return** sonuc;

}

}

\*\*\*\* Deneme 1 \*\*\*\*

{

[ E ?

{

< a ;

}

]

} .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter value :

{

Cümle fonksiyonu

Enter value :

[

If fonksiyonu

If için bir sayı giriniz:

7

Enter value :

?

Koşul doğru

If fonksiyonu koşul bitti

Enter value :

{

If fonksiyonundan cümleye geçiş

Enter value :

<

Çıkış fonksiyonu

Enter value :

a

Çıktı fonksiyonu

Enter value :

;

Çıktı fonksiyonu bitti

Enter value :

}

Cümle fonksiyonu bitti

Enter value :

]

If fonksiyonu bitti

Enter value :

}

Program fonksiyonu bitti

Enter value :

.

Program Bitti

\*\*\*\* Deneme 2 \*\*\*\*

{

{ E ?

{

> a ;

}

}

}.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter value :

{

Cümle fonksiyonu

Enter value :

{

While fonksiyonu

While için bir sayı giriniz:

1

Enter value :

?

Koşul doğru

1

Koşul doğru

2

Koşul doğru

3

Koşul doğru

4

Koşul yanlış

While fonksiyonu koşul bitti

Enter value :

}

Cümle fonksiyonu bitti

Enter value :

}

Program fonksiyonu bitti

Enter value :

.

Program Bitti

\*\*\*\* Deneme 3 \*\*\*\*

{

[ E ?

{

< a ;

}

:

{

> a ;

}

]

} .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter value :

{

Cümle fonksiyonu

Enter value :

[

If fonksiyonu

If için bir sayı giriniz:

5

Enter value :

?

Koşul yanlış

If fonksiyonu koşul bitti

Enter value :

:

If fonksiyonu else mevcut

Enter value :

{

If fonksiyonundan cümleye geçiş

Enter value :

>

Giriş fonksiyonu

Enter value :

a

Girdi fonksiyonu

Enter value :

;

Girdi fonksiyonu bitti

Enter value :

}

Cümle fonksiyonu bitti

Enter value :

]

If fonksiyonu bitti

Enter value :

}

Program fonksiyonu bitti

Enter value :

.

Program Bitti

\*\*\*\* Deneme 4 \*\*\*\*

{

[ E ?

{

= “metin” ;

}

]

} .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter value :

{

Cümle fonksiyonu

Enter value :

[

If fonksiyonu

If için bir sayı giriniz:

6

Enter value :

?

Koşul doğru

If fonksiyonu koşul bitti

Enter value :

{

If fonksiyonundan cümleye geçiş

Enter value :

=

Atama fonksiyonu

Atanacak bir String giriniz:

Ali

Enter value :

;

Atama fonksiyonu bitti

Ali

Enter value :

}

Cümle fonksiyonu bitti

Enter value :

}

Program fonksiyonu bitti

Enter value :

.

Program Bitti