Semàntic

Semàntic 1

- L'ANTLR és un Esquema de Traducció Atribuït
  - Atributs
  - Variables globals
  - Codi en qualsevol lloc de la part dreta de les regles

Semàntic 2: blocs de codi i atributs

- Utilitzarem codi del llenguatge generat (Java)
- Blocs de codi a la classe sintàctica:
  - @header{}: capçalera, per posar-hi *import*
  - @parser::members{}: variables i mètodes de la classe sintàctica
  - @init{} @after{}: inicialitzador i terminador de cada regla
  - Codi a la part dreta de cada regla, entre {}
- Atributs (propietats) dels tokens:
  - text (String): text del token
  - type (int): identificador de tipus de token
  - line (int): línia de codi on es troba
  - pos (int): posició dins la línia on es troba
  - int (int): valor enter (si en té) del token

Semàntic 2: blocs de codi i atributs

- Atributs de la gramàtica d'atributs: es simulen mitjançant variables.
- Cada regla és una funció que pot tenir paràmetres d'entrada, de sortida i variables locals.
- En la definició de la regla, podem assignar el resultat de regles i de tokens a variables amb tipus autodefinit. Podem reutilitzar variables si no hi ha conflicte de tipus (s'assignen amb la mateixa regla)
- Totes les variables que apareixena una regla s'accediran amb '$var'

```
regla [int p1, char p2] returns [boolean p, int t] locals [char i]

@init{ System.out.println("Entrem a la regla 'regla'");}

@after{System.out.println("Sortim de la regla 'regla'");} :

a=regla2[p1] { $p=$a.r; }

b=TKCOMA { System.out.println($b.text + " a la linia " $b.line); }

c=regla3[p2] {$t=$c.r;}

;
```

Detecció d'errors sintàctics

- Podem sobreescriure el mètode de la classe Parser per detectar quan hi ha hagut errors sintàctics:

```
@parser::members{

    boolean errorSintactic=false;

    //Override

    public void notifyErrorListeners(Token offendingToken, String msg,
    RecognitionException e){

        //Si volem conservar el comportament inicial

        super.notifyErrorListeners(offendingToken,msg,e);

        //Codi personalitzat

        errorSintactic=true;

    }

}
```

Semàntic 3: Taula de símbols

- Us proporcionem una Taula de Símbols que conté entrades de tipus Registre
  - SymTable.java
  - Registre.java
- Caldrà fer les modificacions pertinents

Semàntic 4: control semàntic

- Inserir a TS les variables, constants, accions, funcions, tipus, etc. amb les característiques pertinents
- Comprovar que els símbols (vars, funcions, etc.) que accedim existeixin a la TS
- Correctesa de tipus d'expressions
- Correctesa de crides d'accions i funcions
- Altres restriccions imposades pel llenguatge
- ...