

Compiladores

Práctica 6 – Iniciando nuestro compilador/interpreter

Objetivos:

- Conocer los principios de creación de un compilador.
- Conocer el uso de gramáticas.
- Conocer la utilidad de árboles sintácticos.

Árbol sintáctico

A continuación veremos cómo generar un árbol sintáctico.

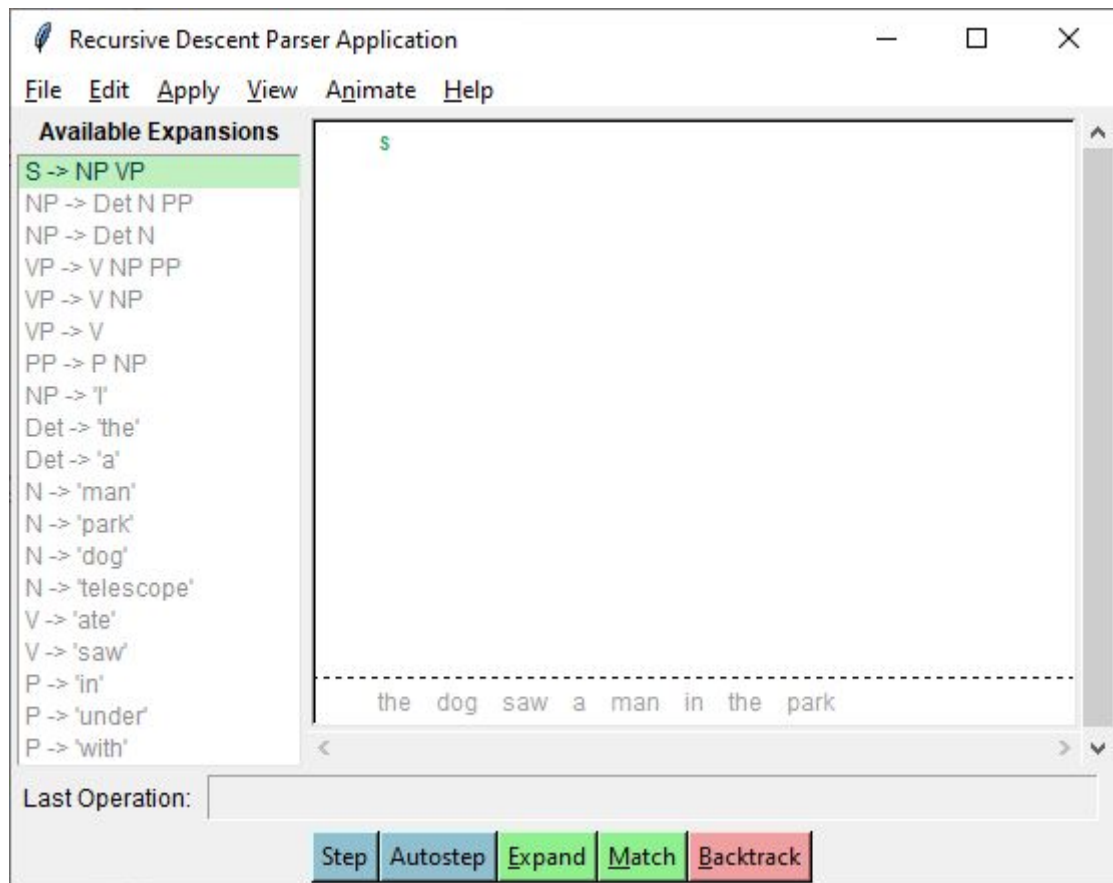
1. Instalar NLTK.

```
pip install nltk
```

2. Escribir los comandos:

```
import nltk  
nltk.app.rdparser()
```

Y deberá aparecer una ventana así:



3. Dar click en **Autostep** y disfruten la animación.

Este es un aplicativo que sirve para visualizar el análisis sintáctico de una manera más amigable. Podemos editar la gramática (Ctrl+g) o el texto (Ctrl+t).

Por ejemplo, intente editar el texto y ver el análisis de las siguientes frases:

- the man saw a dog in the park
- the man ate a dog in the park
- the man saw a man with the dog in the park
- the dog ate a man in the park
- the telescope ate a man with the park
- the man with a man with a man with a man saw a dog with a dog with a dog in the park

¿Notó que las tres últimas frases son un poco extrañas?

¿Quién cree que tiene la culpa? ¿Cómo podríamos resolverlo?

Intente crear una gramática para reconocer:

- La chica estudia compiladores
- El chico estudia compiladores
- La chica toma gaseosa
- El chico toma gaseosa

Sin embargo, la gramática no debe de reconocer inconsistencias como:

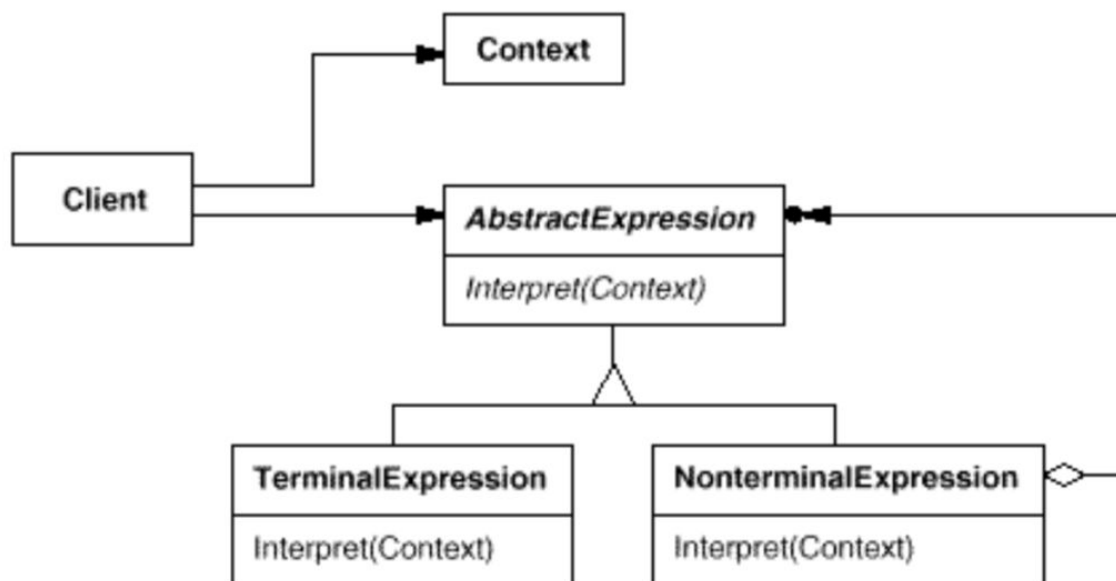
- El chica toma compiladores

Este árbol sintáctico puede ser a veces bastante extenso dependiendo del diseño de la gramática.

Mayor información para el diseño de gramáticas y herramientas útiles:

<https://www.nltk.org/book/ch08.html>

Patrón interpreter



Este es un patrón de diseño que puede reducirse en “interpretate a tí mismo”.

Se caracteriza por heredar de sí mismo y componer a sí mismo.

Cómo funciona puede verse claramente en links de wikipedia:

Ejemplo en Java para un pequeño lenguaje postorden.

[https://es.wikipedia.org/wiki/Interpreter_\(patr%C3%B3n_de_dise%C3%B1o\)](https://es.wikipedia.org/wiki/Interpreter_(patr%C3%B3n_de_dise%C3%B1o))

Ejemplos en otros códigos

https://en.wikipedia.org/wiki/Interpreter_pattern

Ejercicios

Construir una clase que, **a partir de su clase Gramática**, construya en una carpeta las clases:

```
class NodoNoTerminal(AbstractExpressionNT):
    #diccionario<NombreClase, Objeto >
    def interprets(val1,val2,val3):
        return (0,0,0)

...

class Terminal(AbstractExpressionT):
    #valor
    def interprets():
        return valor
```