

T5 – Análise Léxica de Compiladores

Descrição: O trabalho 5 (T5) da disciplina consiste em implementar um gerador de código para a linguagem LA (Linguagem Algorítmica) desenvolvida pelo prof. Jander, no âmbito do DC/UFSCar. O gerador de código deverá produzir código executável em C equivalente ao programa de entrada.

Professor: **Daniel Lucrédio**

Alunos:

- Daniel Watanabe 800697
- Paula Larocca 769705
- Victor Bonometo 800232

Requerimentos:

- Sistema operacional Windows
- Ferramenta IDE APACHE Mavens
- Java JDK

Execução:

1. Inserindo o Antlr4 no Mavens

É necessário colocar no arquivo **pom.xml**:

Java

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>grupo.compiladores</groupId>
  <artifactId>T5</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>
  <dependencies>
    <dependency>
      <groupId>org antlr</groupId>
      <artifactId>antlr4-runtime</artifactId>
      <version>4.13.1</version>
```

```

        </dependency>
    </dependencies>
    <build>
    <plugins>
    <plugin>
        <groupId>org.antlr</groupId>
        <artifactId>antlr4-maven-plugin</artifactId>
        <version>4.13.1</version>
        <configuration>
            <visitor>true</visitor>
        </configuration>
        <executions>
            <execution>
                <id>antlr</id>
                <goals>
                    <goal>antlr4</goal>
                </goals>
            </execution>
        </executions>
    </plugin>
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-jar-plugin</artifactId>
        <version>3.2.0</version>
        <configuration>
            <archive>
                <manifest>
                    <addClasspath>true</addClasspath>
                    <mainClass>grupo.compiladores.t5.T5</mainClass>
                </manifest>
            </archive>
        </configuration>
    </plugin>
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-shade-plugin</artifactId>
        <version>3.2.4</version>
        <executions>
            <execution>
                <phase>package</phase>
                <goals>
                    <goal>shade</goal>
                </goals>
                <configuration>
                    <transformers>
                        <transformer
implementation="org.apache.maven.plugins.shade.resource.ManifestResourceTran
sformer">

```

```

<mainClass>grupo.compiladores.t5.T5</mainClass>
    </transformer>
</transformers>
</configuration>
</execution>
</executions>
</plugin>
</plugins>
</build>
<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>20</maven.compiler.source>
    <maven.compiler.target>20</maven.compiler.target>
    <exec.mainClass>grupo.compiladores.t5.T5</exec.mainClass>
</properties>
</project>

```

E após colocar todos esses plugins e dependências no arquivo **pom.xml**, é necessário colocar a gramática Grama.g4 em um local específico dito pela própria documentação:

Java

```

/NomeProjeto/src/main/antlr4/pacote/Gramatica.g4

//No nosso caso ficou:

/T5/src/main/antlr4/grupo/compiladores/t5/Grama.g4

//Depois basta colocar o arquivo T5.java no

/NomeProjeto/src/main/java/pacote/T5.java

//No nosso caso ficou:

/T5/src/main/java/grupo/compiladores/t5/T5.java

```

E realizar build no projeto, criando o arquivo **T5-1.0-SNAPSHOT.jar**

2. Rodar os casos de testes

Para conseguir rodar o corretor automático deve-se baixar os arquivos necessários dos casos de teste, criar uma pasta temporária para as próprias saídas geradas

Java

```
java -jar
compiladores-corretor-automatico-5.0-SNAPSHOT-jar-with-depen
dencies.jar "java -jar
C:/Users/dahir/OneDrive/Documentos/NetBeansProjects/T5/targe
t/T5-1.0-SNAPSHOT.jar" "C:/MinGW/bin/gcc.exe"
"C:/Users/dahir/OneDrive/Documentos/NetBeansProjects/T5/targ
et/temp"
"C:/Users/dahir/OneDrive/Documentos/NetBeansProjects/T5/targ
et/casostestes" "ra3, ra2, ra1" "t5"
```

Gerando a seguinte saída:

```
=====
Nota do grupo "ra3, ra2, ra1":
CT 1 = 0.0 (0/37)
CT 2 = 0.0 (0/62)
CT 3 = 0.0 (0/9)
CT 4 = 0.0 (0/9)
CT 5 = 10.0 (20/20)
=====
```