Programski Prevodioci 1:

# MikroJava Kompajler

# 

# Opis projekta

Cilj ovog projekta je razvoj funkcionalnog kompajlera za jezik mikro Jave. Projekat je strukturiran u četiri osnovne komponente:

**Lekser**: Ova komponenta se bavi obradom ulaznog izvornog koda i njegovom tokenizacijom. Ukoliko naiđe na nevažeći token, lekser generiše grešku i nastavlja procesiranje. Pravila za validne tokene su definisana u mjlexer.flex fajlu. Tokenizovani izvorni kod se zatim prosleđuje sledećoj komponenti.

**Parser (Sintaksni analizator)**: Parser prima tokenizovan izvorni kod od leksera i vrši sintaksnu analizu, generišući apstraktno sintaksno stablo na osnovu zadate LALR(1) gramatike jezika, koja je definisana u mjparser.cup fajlu.

**Semantički analizator**: Ova komponenta prolazi kroz apstraktno sintaksno stablo (generisano u prethodnoj komponenti) kako bi proverila semantičku ispravnost izvornog koda i generisala tabelu simbola. Za generisanje tabele simbola koristi se symboltable.jar.

**Generator koda**: Nakon završene semantičke analize, ova komponenta ponovo prolazi kroz stablo i koristi generisanu tabelu simbola kako bi generisala bajtkod za izvršno okruženje za Micro Java Virtual Machine (MJVM).

# Opis novih klasa:

1. SemanticAnalyzer.java: Vrši semantičku analizu i generiše tabelu simbola.
2. CodeGenerator.java: Generiše bajtkod za MJVM.
3. CounterVisitor.java: Vrši prebrojavanje promenljivih.
4. DumpSymbolTableVisitorWithBool.java: Proširuje DumpSymbolTableVisitor I dodaje tip Bool
5. TabWithBool.java: Proširuje Tab I dodaje tip Bool
6. RuleVisitor.java: Proširuje VisitorAdaptor
7. Compiler.java: Pokreće kod I generiše obj fajl