|  |  |
| --- | --- |
| etf_logo_2010_cir | **УНИВЕРЗИТЕТ У БЕОГРАДУ**  **ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ**  Програмски преводиоци 1 – 2020/2021.  Булевар краља Александра 73, ПФ 35-54, 11120 Београд, Србија  телефон: 011/3218-321, [dekanat@etf.bg.ac.rs](mailto:dekanat@etf.bg.ac.rs) |

ПРОЈЕКТНИ ЗАДАТАК

***КОМПАЈЛЕР ЗА МИКРОЈАВУ***

Програмски преводиоци 1

Професор: Драган Бојић

Асистенти: Маја Вукасовић,

Кристијан Жижа

Студент:

Мирко Крајцер, 2017/0086

Београд, јануар 2021.

# **О пројектном задатку**

Израда решења пројектног задатка урађена је под оперативним системом Windows у програмском језику Java.

Циљ пројектног задатка је **рализација компајлера за програмски језик Микројаву.**

**Компајлер** омогућава превођење синтаксно и семантички исправних Микројава програма у Микројава бајткод који се извршава на виртуелној машини за Микројаву, уз омогућен опоравак од грешке за предвиђене ситуације.

Програмски преводилац има **четири** основне функционалности: лексичку анализу, синтаксну анализу, семантичку анализу и генерисање кода.

**Лексички анализатор** треба да препознаје језичке лексеме и да врати скуп токена издвојених из изворног кода, који се даље разматрају у оквиру синтаксне анализе.

**Синтаксни анализатор** има задатак да утврди да ли издојени токени могу да формирају граматички исправне сентенце. Генерише апстаракно синтаксно стабло, на основу граматике која је имплементирана у овој фази.

**Семантички анализатор** се формира на основу добијеног апстарактног синтаксног стабла. Спроводи се имплементацијом метода за посећивање чворова у оквиру којих се испитује одговарајућа семантичка правила.

**Генератор кода** преводи синтаксно и семантички исправне програме у извршни облик за одабрано извршно окружење Микројава виртуалне машине.

# **Покретање**

За покретање кода користи се класа Run из библиотеке **mj-runtime-1.1.jar** и као аргумент јој се прослеђује путања до одговарајућег .obj фајла који се тестира.

Да би се генерисао потребан .obj фајл користи се додатно имплементирана класа **MJParserTest.java**.

Пре покретање наведене класе неоходно је покренути **build.xml** који ће уз помоћ алата (**JFlex, AST-CUP**) изгенерисати неопходне јава класе и сместити их на одговарајуће место, које ће уз помоћ осталих, додатно имплементираних класа покренути цео код.

Потпун рад целог програма омогућавају још и библиотеке за лакши испис тј. **log4j,** као и **symboltable-1-1.јаr** која представља коришћену табелу симбола имплементирану преко уланчане листе, за чије претраживање се користи само име симбола, те није могуће имати два истоимена симбола.

# **Тестирање**

У оквиру фолдера тестови се налазе примери тестова.

* Тест1 садржи тестирања у вези семантичке анализе, где је обрађено што више дозвољених конструкција.
* Тест2 обрађује опоравак од грешке за задате случајеве.
* Тест3 представља тест за генерисање кода у којем се налази велики подскуп свих могућих наредби.
* Тест301 представља јавни тест за ниво А.
* Тест302 представља јавни тест за ниво Б.