

Kompilator

Stanisław Woźniak

Kompilator napisany na kurs Języki Formalne i Techniki Translacji.
Gramatyka kompilowanego języka jest opisana w pliku gramatyka.txt .

1 Wymagania

- `g++ -std=c++17`
- GNU Make
- GNU Bison
- Flex

2 Kompilacja

Aby skompilować sam kompilator należy wywołać komendę: **make** będąc w folderze **src/**.
Natomiast aby skompilować potem program kompilatorem należy:

1. plikowi z kodem do kompilacji dać rozszerzenie **.imp** .
2. wywołać: `./compiler <opcje> <plik wejściowy> <plik wyjściowy>` (kolejność argumentów jest dowolna).

2.1 UWAGA!

Plik wyjściowy musi mieć rozszerzenie **.mr**. Jednakże plik wyjściowy nie jest koniecznym argumentem do kompilacji. W przypadku kiedy nie zostanie on podany, kompilator skompiluje plik wejściowy do pliku o tej samej nazwie lecz z rozszerzeniem **.mr** . Przykład: `./compiler plik_0.imp` wygeneruje plik **plik_0.mr**.

2.2 Opcje

Aby kompilator wykrył podawane opcje należy poprzedzić je znakiem **-** (myślnikiem). Można myślnik dać przed każdą opcją z osobna, albo jeden przed wieloma. Jednak wykrywanie opcji po myślniku kończy się kiedy napotkany zostanie znak pusty.

- **s** - (silent) tryb cichy kompilacji nie wyświetli powiadomień typu **WARNING**.
- **d** - (debug) podczas kompilacji wypisze na standardowym wyjściu logów informacje przydatne podczas debuggowania konkretnych fragmentów kodu wewnątrz kompilatora.
- **o** - (optimization) kompilowanie z użyciem optymalizacji (aktualnie ślepa opcja, nie ma żadnego wpływu na działanie kompilatora czy kod wyjściowy).
- **stat** - wypisuje na standardowym wyjściu logów informacje o stworzonych obiektach podczas kompilacji: reprezentacja AST, przypisanie rejestrów, zmienne i stałe oraz postać pośrednia (nazwana pseudo assemblerem).

Przykład:

`./compiler -os plik_1.imp` jest tym samym co: `./compiler -o -s plik_1.imp`