Aynel Gül & Andrea van den Hooff
Assignment 3
Compilerbouw
9 Maart 2018

ORIGINAL GRAMMAR

```
Expr -> ID
        | Expr + Expr
        | -Expr
        | Expr++
        | Expr--
```

Assignment 11

```
Expr ->  Expr + Term
        | Term
Term -> -Term
        | Factor
Factor -> Factor --
        | Factor ++
        | ID
```

Assignment 12

```
Expr -> Term Expr’
Expr’ -> + Expr Expr’
        | epsilon

Term -> Factor Term’
Term’ -> -Term Term’
        | epsilon

Factor -> ID Factor’
Factor’ -> Factor ++
        | Factor --
        | ID
        | epsilon
```

Assignment 13

```
Start -> Expr

Expr -> Term Expr’
Expr’ -> + Expr Expr’
        | epsilon

Term -> Factor Term’
Term’ -> - Term Term’
         | epsilon

Factor -> ID Factor’
Factor’ -> ++ Factor’
        | -- Factor’
        | epsilon
```

Assignment 14

```
Start() { return Expr() && (nextToken() == eof); }

Expr() { return Term() && ExprP(); }

ExprP() {
    token = nextToken();
    Switch(token) case (add) : return Expr() && ExprP();
    default:
        ungetToken(token);
        return true;
}

Term() { return Factor() && TermP(); }

TermP() {
    token = nextToken();
    Switch(token) case (unarymin) : return Term() && TermP();
    default:
        ungetToken(token);
        return true;
}

Factor() { Return(nextToken() == ID) && FactorP(); }

FactorP() {
    token = nextToken();
    Switch(token) case (incr) : return FactorP();
    case (decr):
        return FactorP();
    default:
        ungetToken(token);
        return true;
}
```