

Aufgabe 3: Rekursiver Abstieg (2)

Schreiben Sie Parsermethoden für folgende Produktionen:

```
VarDecl = ("static" | "final") Type ident {" , " ident} ", ".  
Type    = ident .
```

Lösung

```
static void VarDecl() {  
    if (sym == static_) check(static_);  
    else if (sym == final_) check(final_);  
    else error("static or final expected");  
    Type();  
    check(ident);  
    while (sym == comma) {  
        check(comma);  
        check(ident);  
    }  
    check(semicolon);  
}
```

```
static void Type() {  
    check(ident);  
}
```

Die Methode `VarDecl` lässt sich noch optimieren, indem `check` durch `scan` ersetzt wird, wenn vorher bereits geprüft wurde, dass das richtige Symbol vorliegt:

```
static void VarDecl() {  
    if (sym == static_) scan();  
    else if (sym == final_) scan();  
    else error("static or final expected");  
    Type();  
    check(ident);  
    while (sym == comma) {  
        scan();  
        check(ident);  
    }  
    check(semicolon);  
}
```