

Codeerzeugung – Teil 1

(24 Punkte)

Erweitern Sie Ihren Compiler um Codeerzeugung gemäß der *MicroJava*-VM-Spezifikation, siehe VO-Skriptum Kapitel 5, Seiten 12-16. Die dafür nötigen Klassen *Code* und *Operand* befinden sich im Package *ssw.mj.codegen*.

Implementieren Sie in der Klasse *Code* die Methoden zur Ausgabe häufig verwendeter Code-Muster, zB: Laden von Werten sowie Zuweisungen. Vervollständigen Sie die Klasse *Struct* um die Methoden zur Typ-Kompatibilitäts-Prüfung, siehe VO-Skriptum, Kapitel 4.3, Seite 8. Die Klasse *Operand* ist bereits vollständig implementiert. Die Klasse *Label* benötigen Sie erst für die nächste Übung.

Zusätzlich befindet sich im Paket ein Decoder (*Decoder.java*), der Objektcode der *MicroJava*-VM in textueller Form ausgibt. Sie können ihn als Hilfsmittel verwenden, um Ihren erzeugten Objektcode zu überprüfen. Den generierten Code aus den JUnit-Testfällen können Sie ausgeben lassen wenn Sie das Feld *Configuration.PRINT_DECODED_BYTE_CODE* auf *true* setzen.

In diesem Teil der Codeerzeugung sollen Sie *nur* die Teile implementieren, die im VO-Skript bis einschließlich Folie 6.54 (Zuweisungen) beschrieben sind. Zusätzlich müssen für Methoden die Codes *enter*, *exit* und *return* ausgegeben werden. Folgende Felder im Codebuffer müssen gesetzt sein: *dataSize*, *mainpc*.

Den Code für Sprünge und Methoden-Aufrufe müssen Sie erst in Übung 6 implementieren.

Denken Sie daran, alle Kontextbedingungen, die Sie nun prüfen können, auch tatsächlich zu prüfen und entsprechende Fehlermeldungen auszugeben.

Abgabe und Hinweise

Die Abgabe der Übungen muss elektronisch erfolgen. Geben Sie folgende Dateien ab:

- Elektronisch in das Repository: Alle Quellcode-Dateien, die zum Ausführen des Compilers benötigt werden (Packages *ssw.mj*, *ssw.mj.codegen* und *ssw.mj.symtab*), also auch alle Klassen der Angabe. Die Verzeichnis-Struktur muss erhalten bleiben.
- `svn://ssw.jku.at/2015W/UB/k<MatrNr>/branches/UE5`

JUnit Testfälle: *ScannerTest*, *ParserTest*, *SymbolTableTest*, *SimpleCodeGenerationTest*