

Script generated by TTT

Title: Petter: Compiler Construction (02.07.2020)
- 50: Registers

Date: Wed Jul 01 14:43:03 CEST 2020

Duration: 09:48 min

Pages: 8

Chapter 2: Generating Code for the Register C-Machine

8/49

Simple Expressions and Assignments in R-CMa

Task: evaluate the expression $(1 + 7) * 3$
that is, generate an instruction sequence that

- computes the value of the expression and
- keeps its value accessible in a reproducible way

9/49

Simple Expressions and Assignments in R-CMa

Task: evaluate the expression $(1 + 7) * 3$
that is, generate an instruction sequence that

- computes the value of the expression and
- keeps its value accessible in a reproducible way

Idea:

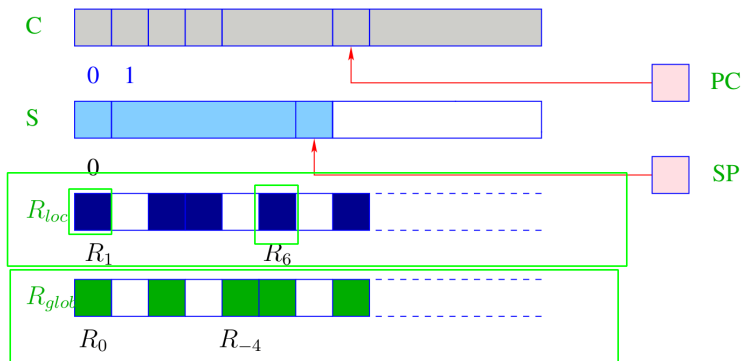
- first compute the value of the sub-expressions
- store the intermediate result in a temporary register
- apply the operator
- loop

9/49

Principles of the R-CMa

The R-CMa is composed of a stack, heap and a code segment, just like the JVM; it additionally has register sets:

- **local registers** are $R_1, R_2, \dots, R_i, \dots$
- **global registers** are $R_0, R_{-1}, \dots, R_j, \dots$



10/49

The Register Sets of the R-CMa

The two register sets have the following purpose:

- 1 the **local** registers R_i
 - save temporary results
 - store the contents of local variables of a function
 - can efficiently be stored and restored from the stack
- 2 the **global** registers R_i
 - save the parameters of a function
 - store the result of a function

11/49

The Register Sets of the R-CMa

The two register sets have the following purpose:

- 1 the **local** registers R_i
 - save temporary results
 - store the contents of local variables of a function
 - can efficiently be stored and restored from the stack

11/49

The Register Sets of the R-CMa

The two register sets have the following purpose:

- 1 the **local** registers R_i
 - save temporary results
 - store the contents of local variables of a function
 - can efficiently be stored and restored from the stack
- 2 the **global** registers R_i
 - save the parameters of a function
 - store the result of a function

Note:

for now, we only use registers to store temporary computations

11/49

The Register Sets of the R-CMa

The two register sets have the following purpose:

- 1 the *local* registers R_i
 - save temporary results
 - store the contents of local variables of a function
 - can efficiently be stored and restored from the stack
- 2 the *global* registers R_i
 - save the parameters of a function
 - store the result of a function

Note:

for now, we only use registers to store temporary computations

Idea for the translation: use a register counter i

- registers R_j with $j < i$ are *in use*
- registers R_j with $j \geq i$ are *available*