EXPRESSIONS

Tomáš Korgo, Lukáš Hudec

Expressions

- What are expressions?

  - combination of values, constants, variables and operators

  - they compute and return another value

- Examples:

  2+5, y+9, a*b*c ...

Expressions

- How are they evaluated?

  - evaluated one operator at a time

  - precedence and associativity are detremining order of evaluation

Temporary-variable allocation

- Why use temporary variables?

  - operation is creating a temporary variable that references the result

  - this temporary is used as operand at the next evaluation stage

Temporary-variable allocation

- Back end
- Registry
- Static memory
- Stack

Temporary-variable allocation

- Back end

  - temporary variable type encoded in the name

Temporary-variable allocation

- Registry

  - can be allocated using a stack of register names

  - pseudo registers (run-time variables)

- Very fast
- Problems with functions

Temporary-variable allocation

- On Stack

  - region of stack frames

- Maximum region size varies

  - controlled by the worst-case expression subroutine

Rvalues and Lvalues

- Rvalues
  - hold value
  - can go the the right of the equal sign

Input	Code

x+y	t0 = x;
	t1 = y;
	t1 += t0;

Rvalues and Lvalues

- Lvalues

  - hold adress

  - generated by subexpressions to the left of an equal sign

- Logical Lvalues – evaluate to the adress
- Physical Lvalues – hold adress