

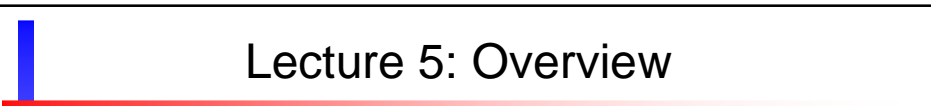
# EECS 10: Computational Methods in Electrical and Computer Engineering

## Lecture 5

Rainer Dömer

doemer@uci.edu

The Henry Samueli School of Engineering  
Electrical Engineering and Computer Science  
University of California, Irvine



## Lecture 5: Overview

- Basic Types in C
- Arithmetic Operations in C

## Basic Types in C

- Integer types
  - **char** Character, e.g. `'a'`, `'b'`, `'1'`, `'*'`
    - typical range [-128,127]
  - **short int** Short integer, e.g. -7, 0, 42
    - typical range [-32768,32767]
  - **int** Integer, e.g. -7, 0, 42
    - typical range [-2147483648,2147483647]
  - **long int** Long integer, e.g. -991, 91, 1231
    - typical range [-2147483648,2147483647]
  - **long long int** Very long integer, e.g. 1234511
    - typical range [-9223372036854775808, 9223372036854775807]
- Integer types can be
  - **signed** negative and positive values (and 0)
  - **unsigned** positive values only (and 0)

EECS10: Computational Methods in ECE, Lecture 5

(c) 2004 R. Doemer

3

## Basic Types in C

- Floating point types
  - **float** Floating point with single precision
    - Example 3.5f, -0.234f, 10E8f
  - **double** Floating point with double precision
    - Example 3.5, -0.23456789012, 10E88
  - **long double** Floating point with high precision
    - Example 12345678.123456E1231
- Floating point values are in many cases *approximations* only!
  - Storage size of floating point values is fixed
  - Many values can only be represented as approximations
  - Example:  $1/3 = .333\dots$

EECS10: Computational Methods in ECE, Lecture 5

(c) 2004 R. Doemer

4

## Conversion Specifiers for Basic Types

Type	printf()	scanf()
• long double	%Lf	%Lf
• double	%f	%lf
• float	%f	%f
• unsigned long long	%llu	%llu
• long long	%lld	%lld
• unsigned long	%lu	%lu
• long	%ld	%ld
• unsigned int	%u	%u
• int	%d	%d
• short	%hd	%hd
• char	%c	%c

EECS10: Computational Methods in ECE, Lecture 5

(c) 2004 R. Doemer

5

## Arithmetic Operations in C

- Arithmetic Operators
  - parentheses ( , )
  - unary plus, minus +, -
  - multiplication, division, modulo \*, /, %
  - addition, subtraction +, -
  - shift left, shift right <<, >>
- Evaluation order of expressions
  - usually left to right
  - by operator precedence
    - see table above!
- Arithmetic operators are available
  - for integer types: all
  - for floating point types: all except %, <<, >>

EECS10: Computational Methods in ECE, Lecture 5

(c) 2004 R. Doemer

6