

## Assorted Announcements

at end of lecture (worth 5 points). Be sure to sign in on  
s circulating.

next Tuesday (7 May).

ments due Friday, 10 May.

esday, 15 May at 7PM in 2040 VLSB.

# Course Summary

Languages

of Programming Languages

of Complex Software

# Programming Languages

declarations

scope (lifetime) of variables

relationship between language design and runtime structures:

memory representation

support for recursion, variable-sized data, functional values

performance

single vs. multiple inheritance

style interfaces

methods for describing languages: type systems

languages used here: Prolog, Python, C++.

# Translation of Programming Languages

Analysis

Expressions, finite automata

Regular syntax

LR, recursive descent

LR, shift-reduce parsing

LR: derivation

LR: driven translation

Semantics

LR: tables, relation to environment diagrams

LR: type inference

## tion of Programming Languages, contd.

tion, intermediate forms

representations for "special effects"

s

e calls

oriented method dispatch

collection

n

gy: basic blocks, control-flow graph

' optimizations

e of flow analysis

## Tools

ation, use of regular expressions and states

ators, rule-based programming

rol concepts

## Construction of Complex Software

with project, including parts you didn't write.

"pass" or "phase".

Task-orientation to partition task

Use of intermediate forms; how used to reduce work of  
programmers

## Parting Remarks

compilers:

this course are general-purpose tools

main-specific languages

es for research

n and distributed computation

rogram analysis:

ts compiling for parallelism & distributed computation.

ng programs for security attacks/flaws

analysis for program validation (e.g., avionics)