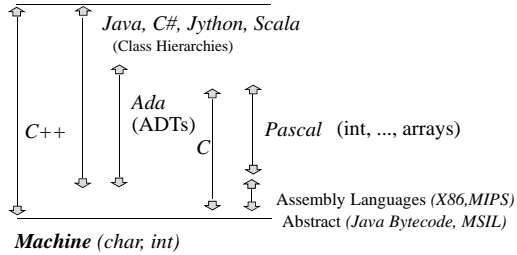


Levels of Abstraction

Problem Domain (*stacks, tables*)



Machine (*char, int*)

cs480(Prasad)

L0Intro

1

What to expect from the course?

◆ **Superficially** : Features in *Java, C#, C++, Scheme, Scala*

◆ **Broader Perspective** :

Paradigms : Imperative, Functional, Object-oriented
Concepts :

- » Abstract Data Types; Encapsulation
- » Features for Reuse
 - Class hierarchy; Polymorphism
- » Features for Reliability
 - Strong typing
 - Exception mechanism
- » Recursion and List Processing.

cs480(Prasad)

L0Intro

2

Related Languages

Java

↑
C++, C#
Scala, Jython
Modula-2, Modula-3, Oberon
Eiffel, Ada-95

Scheme

↑
LISP, Common LISP
ML
Haskell

cs480(Prasad)

L0Intro

3

Example: Simple Language Design Issue

□ Reserving keywords contributes to simplicity.

```
IF IF = THEN THEN THEN=ELSE ;  
(Confusing but legal in PL/I)
```

□ Control Abstraction

```
if C then S1 else S2  
vs  
if C goto 1 ;  
S2 ;  
goto 2 ; (* FORTRAN *)
```

```
1 : S1 ;
```

```
2 :
```

Evolution of Programming Languages

- **FORTRAN** (FORMula TRANslator)

Goals : Scientific Computations
Efficiency of execution
Compile-time storage determination

Features : Symbolic Expressions
Subprograms
Absence of Recursion

(John Backus : 1977 Turing Award)

- **COBOL**

Goal: Business Application
Features : Data Definition and File Handling

(Grace Murray Hopper)

cs480(Prasad)

L0Intro

5

Evolution of Programming Languages

- **ALGOL- 60** (ALGOrithmic Language)

Goals : Communicating Algorithms
Features : Block Structure (Top-down design)
Recursion (Problem-solving strategy)
BNF - Specification

(Peter Naur : 2005 Turing Award)

- **LISP** (LISt Processing)

Goals : Manipulating symbolic information
Features : List Primitives
Interpreters / Environment

(John McCarthy : 1971 Turing Award)

cs480(Prasad)

L0Intro

6

C.A.R Hoare On Algol-60

- ◆ Here is a language so far ahead of its time, that it was not only an improvement on its predecessors, but also on nearly all its successors.
- ◆ I conclude that there are two ways of constructing a software design: One way is to make it so simple that there are obviously no deficiencies and the other way is to make it so complicated that there are no obvious deficiencies.

(C. A. R. Hoare : 1980 Turing Award)

cs480(Prasad)

L0Intro

7

Evolution of Programming Languages

- **PL / 1**

FORTRAN + COBOL + SNOBOL+
... + concurrency + ...

“ When FORTRAN has been called infantile disorder, full PL /1, with its growth characteristics of a dangerous tumor, could turn out to be a fatal disease.”

---- **E. W. Dijkstra**
(1972 Turing Award Lecture)

cs480(Prasad)

L0Intro

8

Evolution of Programming Languages

- **SIMULA** (SIMULAtion LAnguage)

Features : Data Abstraction .
Class Hierarchies. (Inheritance)
(O. J. Dahl, K. Nygaard : 2001 Turing Award)

- **C**

Goal : Systems Programming

Features : Coding language for Unix. Portability.
(D. Ritchie and K. Thompson : 1983 Turing Award)

cs480(Prasad)

L0Intro

9

On C and C++

◆ C makes it easy to shoot yourself in the foot, C++ makes it harder, but when you do, it blows away your whole leg. -- Bjarne Stroustrup

◆ The last good thing written in C was Franz Schubert's Symphony number 9.

◆ C is quirky, flawed, and an enormous success.
-- Dennis M. Ritchie.

cs480(Prasad)

L0Intro

10

Evolution of Programming Languages

- **Pascal**

Goal : Structured Programming, Compiler writing.

Features :

- Rich set of data types for efficient algorithm design
 - E.g., Records, sets, ...
- Variety of "readable" *single-entry* *single-exit* control structures
 - E.g., *for-loop*, *while-loop*,...
- Efficient Implementation
 - Recursive descent parsing

(N. Wirth : 1984 Turing Award)

cs480(Prasad)

L0Intro

11

On Type System; Efficiency

◆ Type security is intended not so much to inspire programmers as to protect them from their own not inconsiderable frailties.

◆ More computing sins are committed in the name of efficiency (without necessarily achieving it) than for any other single reason - including blind stupidity.

-- William A. Wulf

cs480(Prasad)

L0Intro

12

Other Languages

- ◆ Functional
 - » Common LISP, Scheme
 - » ML, Haskell
 - (Robin Milner : 1991 Turing Award (for ML))
- ◆ Logic
 - » Prolog
- ◆ Object-oriented
 - » Smalltalk, Eiffel, Java, C#
 - (Alan Kay : 2003 Turing Award (for SmallTalk))
 - » C++, Ada-95, CLU
 - (Barbara Liskov: 2009 Turing Award (for CLU))
 - » Modula-3, Oberon
- ◆ Application specific languages and tools

cs480(Prasad)

L0Intro

13

Modern Scripting Languages

- ◆ Multiparadigm Constructs =
OOP style + Functional style
(+ Ease of prototyping (Interpreter-based))
- ◆ *Examples:* Python, Ruby, PERL, PHP, ...,
JPython, JRuby, ..., CAML, F#
Scala, ...

cs480(Prasad)

L0Intro

14

On Comparing Languages

- ◆ I have reaffirmed a long-standing and strongly held view: Language comparisons are rarely meaningful and even less often fair. A good comparison of major programming languages requires more effort than most people are willing to spend, experience in a wide range of application areas, a rigid maintenance of a detached and impartial point of view, and a sense of fairness.
 - ◆ Bjarne Stroustrup, [The Design and Evolution of C++](#)

cs480(Prasad)

L0Intro

15