


# CSE 3302 Programming Languages

## History of Programming Languages

Chengkai Li  
Fall 2007

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 1

## Languages you used or heard about



Ada	Prolog	MATLAB
Basic	Python	Progress
C	Ruby	ThoroughBred
C++	Scheme	COBOL
C#	Smalltalk	FORTH
Fortran		Delphi
Haskell		JavaScript
Java		SQL
Lisp		LOGO
ML		B
Pascal		Large
Perl		J#
PHP		Postscript
		XML
		HTML
		ModPerl

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 2

## Can anybody sing this song?



- **99 Bottles of Beer**  
99 bottles of beer on the wall, 99 bottles of beer.  
Take one down and pass it around, 98 bottles of beer on the wall.

98 bottles of beer on the wall, 98 bottles of beer.  
Take one down and pass it around, 97 bottles of beer on the wall.


...

1 bottle of beer on the wall, 1 bottle of beer.  
Take one down and pass it around, no more bottles of beer on the wall.

No more bottles of beer on the wall, no more bottles of beer.  
Go to the store and buy some more, 99 bottles of beer on the wall.

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 3


## C



```
static void fill_buffer(char *buf, int b) {
    char line[BUFSIZ/2];
    if (b>0) {
        sprintf(buf, "%d bottle%s of beer on the wall, %d bottle%s of beer.\n" \
            "Take one down and pass it around, ", b, PLURALS(b), b, PLURALS(b));
        if (b==1)
            strcat(buf, "no more bottles of beer on the wall.\n");
        else {
            sprintf(line, "%d bottle%s of beer on the wall.\n", b-1, PLURALS(b-1));
            strcat(buf, line);
        }
    } else {
        sprintf(buf, "No more bottles of beer on the wall, no more bottles of beer.\n" \
            "Go to the store and buy some more, 99 bottles of beer on the wall.\n");
    }
}
....
```

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 4

## Prolog




```
report_bottles(0) :- write('no more bottles of beer'), !.
report_bottles(X) :- write(X), write(' bottle'),
    (X = 1 -> true ; write('s')),
    write(' of beer').

report_wall(0, FirstLine) :-
    (FirstLine = true -> write('No '); write('no ')),
    report_bottles('more'), write(' on the wall'), !.
report_wall(X, _) :- report_bottles(X), write(' on the wall').

...
```

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 5

## Assembly



```
inc    eax            ; get the original value
push  eax            ; convert it to string
lea   eax, [ebp - 04h]
push  eax            ; string will be stored here
call  _integer_to_string
lea   eax, [ebp - 04h]
push  eax
call  _show_line     ; 'xx'
push  _line_1_1
call  _show_line     ; ' bottles of beer on the wall, '
lea   eax, [ebp - 04h]
push  eax
...
```

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 6



## Thousands out there

- [http://en.wikipedia.org/wiki/Timeline\\_of\\_programming\\_languages](http://en.wikipedia.org/wiki/Timeline_of_programming_languages)
- [http://www.oreilly.com/news/graphics/prog\\_lang\\_poster.pdf](http://www.oreilly.com/news/graphics/prog_lang_poster.pdf)

Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      13

## The most popular PLs?

- Popularity:
  - Most widely used?
  - Most lines of codes?
  - Most jobs?
  - Most courses/projects?
  - Most search engine queries?
- [http://www.welton.it/articles/language\\_popularity.html](http://www.welton.it/articles/language_popularity.html)
- <http://www.developer.com/lang/article.php/3390001>

Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      14



## History

- Early History : The First Programmer
- The 1940s: The First Computers
- The 1950s: The First Programming Languages
- The 1960s: An Explosion in Programming Languages
- The 1970s: Simplicity, Abstraction, Study
- The 1980s: New Directions and OO
- The 1990s: Consolidation, Internet, Libraries, and Scripting
- The Future

Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      15

## The First Programmer

- Before the birth of computers
- Jacquard loom (early 1800s)
  - translated card patterns into cloth designs.
  - <http://www.youtube.com/watch?v=Sn337QHU0AY>






http://www.smith.edu/hsc/silk/History/jacquard.html  
www.mdb.com/people/799/000097508/

Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      16

## The First Programmer

- Charles Babbage's analytical engine (1830s and 1840s)
  - Devoted entirely to computation
  - Programs: cards with data and operations
  - Difference Engine: which inspired Analytical Engine (the design was realized in 1991)
  - [http://www.youtube.com/watch?v=KL\\_wy-CxBP8](http://www.youtube.com/watch?v=KL_wy-CxBP8)
- Ada Lovelace – first programmer (daughter of Byron)


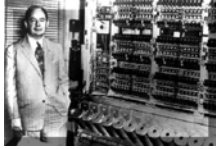



http://en.wikipedia.org/wiki/Charles\_Babbage      http://en.wikipedia.org/wiki/Ada\_Lovelace

Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      17

## The First Computers

- ENIAC (1943)
  - First electronic computer
  - U. Penn
- EDVAC (1945)
  - John von Neumann
  - von Neumann architecture
  - “Stored program”: data and programs in the same space





http://en.wikipedia.org/wiki/ENIAC      www.agers.cfwb.be/apsdx/figinfo26.htm


Lecture 2 - History, Fall 2007      CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007      18

## The First Computers

- **Z3 (1941)**
  - Konrad Zuse
  - First digital computer
  - Electromechanical, rather than electronic
- **Plankalkul (Plan Calculus) : (1945)**
  - Eventually published in 1972
  - First compiler implemented in 2000



irb.cs.tu-berlin.de/.../de/Konrad\_Zuse\_2.html



http://en.wikipedia.org/wiki/Konrad\_Zuse

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 19

## Machine Codes and Assembly Language

- **Machine code:** bit sequences

```
000000 00001 00010 00110 00000 100000
100011 00011 01000 00000 00001 000100
000010 00000 00000 00000 10000 000001
```

- **Assembly program:** symbolic representation of machine codes

```
LDA SUB
CMA
INC
ADD MIN
STA DIF
```

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 20

## The 1950s: The First Programming Languages

- **FORTRAN:** the first higher-level programming language

Languages following FORTRAN

- COBOL
- Algol60
- LISP
- APL

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 21

## FORTRAN

- The first language  
1954-1957  
John Backus, et. al. (IBM)
- Scientific and engineering applications (FORmula TRANslation).
- Goal: generate fast machine code. Its compiler is still among the most efficient.
- Contributions: array, loops by indexed variables, if-statement
- Still widely used today (Fortran, II, III, IV, 66, 77, 90, 95, 2003, 2008).


Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 22

## FORTRAN

- John Backus: IBM group

1977 ACM Turing Award: “for profound, influential, and lasting contributions to the design of practical high-level programming systems, notably through his work on FORTRAN, and for seminal publication of formal procedures for the specification of programming languages.”

<http://www.youtube.com/watch?v=xQtT2sRkOEw>



www.columbia.edu/acis/history/backus.html

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 23

## Major languages following FORTRAN

- COBOL
- Algol60
- LISP
- APL

Lecture 2 - History, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 24

## COBOL



- Common Business-Oriented Language  
1959-1960  
Grace Hopper, et. al. (US Department of Defense)
- Business applications: banks and corporations
- Still widely used



[http://en.wikipedia.org/wiki/Grace\\_Murray\\_Hopper](http://en.wikipedia.org/wiki/Grace_Murray_Hopper)

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

25

## COBOL



- Goal: allow nonprogrammers to read/understand programs
- Consequences:
  - Very wordy, like English
    - C++ vs. *ADD 1 TO COBOL GIVING COBOL*
    - Can be difficult to write complex algorithms
  - Human readability improved, or only complicated?
- Contributions:
  - Record structure
  - Separate data structures from execution
  - Output formatting by examples
- COBOL 2002: OO programming

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

26

## Algol60



- ALGOrithmic Language, 1958-1960  
a committee of European and American computer scientists (John Backus and John McCarthy involved)
- Contributions:
  - free-format (which modern language is not free-format?)
  - Backus-Naur forms (BNF) for defining syntax
  - type declarations for variables,
  - block-structure, begin-end
  - recursion,
  - pass-by-value parameters
- Impacts:
  - one of the most influential programming languages
  - most imperative languages are derivatives of Algol: Pascal, C/C++, Ada, Java.
  - standard way of describing algorithms in research papers for 30 years.

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

27

## LISP



- LISP Processor  
late 1950s  
John McCarthy (MIT, at Stanford now)  
1971 Turing Award for contributions in AI.
- AI applications, still dominating
- Contributions:
  - first one to depart from imperative/procedural paradigm: functional programming language
  - Garbage collection
  - Recursion, s-expression
- Limitations:
  - Could not run efficiently on von Neumann architecture
  - LISP-specific machines
- Variants: Common LISP, Scheme
- Following LISP: ML, Haskell

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

28

## APL



- A Programming Language  
late 1950s to early 60s  
Kenneth E. Iverson (Harvard and IBM)
- 1979 Turing Award for contributions to mathematical notation and PL theory.
- A language for programming mathematical computations
  - arrays and matrices
- Functional style, influenced FP and modern function languages
- Drawbacks:
  - No structuring
  - Greek symbol, requires special terminal keyboard
  - Extremely difficult to read

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

29

## Summary of 1950s



- Huge success and big impacts:
  - Pioneered imperative and functional programming
  - Still used much today
  - Many derivatives
- The 1960s is not equally fruitful

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

30

## The 1960s: An Explosion in Programming Languages

- Hundreds of programming languages
- PL/I
- Algol 68
- SNOBOL
- Simula67
- BASIC

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

31

## PL/I

- 1963-1964, IBM
- Goal:
  - Universal language, “language to end all languages.”
    - combine features of FORTRAN, COBOL and Algol60
    - concurrency
    - exception handling
  - for IBM 360
- Can be considered to be a failure:
  - translators were difficult to write, slow, huge and unreliable
  - difficult to learn and use
  - forward-looking, but simply ahead of its time

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

32

## Simula67

- 1965-1967  
Kristen Nygaard and Ole-Johan Dahl (Norwegian Computing Center)  
2001 Turing Award for OO and Simula
- Based on Simula I and Algol60
- Designed for simulations
- First OO language
  - object, class, subclass (inheritance), virtual method, coroutine
- Ahead of its time. Inefficient.

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

33

## Basic

- Beginner's All-purpose Symbolic Instruction Code  
1964  
John Kemeny and Thomas Kurtz (Dartmouth)
- Goal:
  - Simple language for non-experts to use
- Popular for schools and homes
  - Altair BASIC for personal computers, by Bill Gates, Paul Allen, and Monte Davidoff. (1975)
- Dialect: Visual Basic

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

34

## The 1970s: Simplicity, Abstraction, Study

- Tremendous success
  - few new concepts
  - simplicity and consistency
- Algol-W  
Niklaus Wirth (ETH Zurich) (1984 Turing Award) and C.A.R. Hoare (1980 Turing Award)
  - response to the direction in 1960s
- Pascal, 1971, Niklaus Wirth
  - popular for teaching PL
- C, 1972, Dennis Ritchie (Bell Labs), 1983 Turing Award
  - Successful partially due to the popularity of UNIX
- CLU, Euclid, Mesa: Abstract Data Type (ADT)

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

35

## The 80s: New Directions and OO

- Following the experiments of ADT in 70s
  - Ada, Modula-2
- Object-Oriented Programming
  - Smalltalk, C++, Eiffel, Object C, Object Pascal, Oberon
- Functional Programming
  - Scheme, ML, Haskell, Miranda, FP
- Logic Programming
  - Prolog

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

36

## Ada



- Named after Ada Lovelace  
1980  
Department of Defense
  - Hundreds of languages were used by DoD
  - Required in DoD projects, 1987-1997.
- Contributions:
  - ADT (package)
  - concurrency (task)
  - exception handling
- Universal language, PL/I of 80s, but didn't fail
  - carefully designed
  - required use

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

37

## Smalltalk



- 1980  
Alan Kay (2003 Turing Award), Dan Ingalls, et. al. (Xerox PARC)
- Inspired by Simula67
- Contributions
  - purest OO language
  - graphical user interface, mouse (limited its use, as such hardware was not generally available)
  - Push C++ and OO into spotlight
- Still used much today

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

38

## C++



- 1980  
Bjarne Stroustrup (Bell Labs, now at TAMU)
- Extensions from Simula67 and C, "C with Classes"

[C++0x – An Overview](#)

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

39

## Other Paradigms



- Functional Programming:
  - Common Lisp, Scheme, ML, Haskell (pure functional programming language)
  - Logo: teach kids to program  
<http://www.youtube.com/watch?v=ohgPmdZgUmE>
- Logic Programming:
  - Prolog

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

40

## 1990's: Internet, Scripting



- OO widely adopted (C++ was going to dominate)
- Then Java came
- Scripting languages, became general-purpose languages:  
Perl, Tcl, Python, PHP, ...  
<http://www.youtube.com/watch?v=wVYsINZ5nAY>

Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

41

## Java



- 1995, James Gosling et. al. (Sun)
- Was for embedded consumer-electronic applications (set-top box), then for Internet/Web and network applications
- Based on C++
- Differences
  - Pros: richer libraries (API), portability (compile-once, run-anywhere)
  - Cons: slower than C++, no ISO/ANSI standard (controlled by Sun).
  - references vs. pointers
  - garbage collection



Lecture 2 - History, Fall 2007

CSE3302 Programming Languages, UT-Arlington  
©Chengkai Li, 2007

42

## What's next?



- C/C++/C#?
- Java?
- A new language?