



CSE 3302 Programming Languages

Chengkai Li
Fall 2007

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

- What do they have in common?
blue, orange and white


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



Myself: Chengkai Li

- China.
- Research interests:
databases, Web, information retrieval, data mining
- <http://ranger.uta.edu/~cli>
- Looking for undergraduate/graduate students.
Financial supports available.


Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 3



Now it's your turn...

- name, year, where from
- courses taken
- programming skills
- what do you want to get from the course
- anything else


Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 4



Course Page

- <http://crystal.uta.edu/~cli/cse3302>
Syllabus, Announcements, Schedule (lecture notes), Homeworks, Machine Problems, Exams, Discussion Groups, Resources, Accommodation based on disability.
- Important announcements:
announcement page
(<http://crystal.uta.edu/~cli/cse3302/announcements.htm>)
- Web CT
 - discussion group.
 - assignments submissions.
- **Tentative!**

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 5



Basics

- Lectures: Tuesday / Thursday , 3:30-4:50pm, NH 110
– Slides don't cover everything. You need to read the book and supplementary materials.
- Instructor: Chengkai Li
Office hours: Wed./Fri. 11am-12pm , [Nedderman Hall 334](#)
Contact: (817) 272-0162, cli@uta.edu
- TA: Roochi Mishra
Office hours: Tu/Th 2-3pm [Nedderman Hall 232](#)
Contact: (817) 272-5188, roochimishra@uta.edu
- Textbook: *Programming Languages - Principles and Practices*, 2nd Edition, by Kenneth C. Loudon.
- Reference book: *Principles of programming languages: design, evaluation, and implementation*, 2nd Edition, by Bruce J. MacLennan.

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 666

Basics (cont.)



- **Prerequisites**

- *Background:*

CSE 3315 Theoretical Concepts in Computer Science and Engineering.

Automata, grammar, regular expression, ...

- *Programming:*

Experiences with at least one major programming language.

Tentative grading scheme



- Homeworks (HW): 10%, individual.
- Machine Problems (MP): 30%, individual or group of 2 (no exception).
- Essays (ES): 10%, individual.
- Midterm exam: 20%, close-book, close-notes.
- Final exam: 30%, close-book, close-notes.
- Bonus points: 5%, based on class participation.

- **Out:** Thursdays
- **In (Due):** Thursdays 5pm (3 weeks later)
By **WebCT**. (Will do Email if we cannot setup WebCT.)

Homeworks (HW) 10%



- HW1, HW2
- Problems based
- Can discuss, but must write it independently.
- Review the materials learned
- Get you prepared for the exams

Machine Problems (MP) 30%



- MP1, MP2, MP3.
- Can discuss, but must do it independently or in group of 2 (must roughly equally contribute).
- Decent amount of programming:
 - Not crazy.
 - But not a piece of cake either.

Essays (ES) 10%



- ES1, ES2
- satisfying ABET requirements
- Can discuss, but must write it independently.
- ES1: some extra readings.
- ES2: preparations for panel discussion.

Deadlines



- Automatic 24-hour grace period, without penalty.
- 0 after the grace period.
- Example:
 - HW1.due = Oct. 4th, 5pm
 - If (you submit before Oct. 5th 5pm) then
you are fine
 - else
HW1.score = 0

Exam



- **Midterm: 20%**
Oct. 11th Review
Oct. 16th Exam (in class), close-book, close-notes (may become take-home essay)
- **Final: 30%**
Dec. 6th Review
Dec. 13th Exam, Thursday, 2-4:30pm, close-book, close-notes

Regrade



- 7 days after we sent out scores (either by Web CT or email). TA will handle regrade requests. Won't consider it after 7 days.
- If not satisfied with the results, 7 days to request again. Instructor will handle it, and the decision is final.

How to get bonus points? 5%



- **During lectures:**
 - Raise questions. (There are no bad questions.)
 - Actively participate in discussions.
 - But don't overdo it please.
- **Presentation and Panel Discussions:**
 - Students form groups. Each group study a language not covered in lectures. And try it out.
 - Each student writes an essay that involves the languages studied by other groups.
 - The group presents (either single or multi speakers is fine.)
 - The group discusses as a panel. Instructor moderates. Questions from other students.

Get questions after class?



- **Discuss with classmates:**
 - The best way to learn is to teach.
- **General questions (that can be asked by any student):**
 - Post on [discussion group](#) (Web CT or listserv)
- **Individual questions (that apply to you specifically):**
 - Email TA
 - No need to CC me. If TA cannot handle it, she will forward to me.
- **Come to TA's office hours or my office hours.**

Get bored




- Do you watch Youtube?

<http://www.youtube.com/watch?v=qC2ew6qLa8U>


<http://www.youtube.com/watch?v=463gKcXDVzQ>

Don't do it. It's not worth it.



read & sign the statement


Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 19



CSE 3302: What is it about?

- Exposure to various kinds of languages and paradigms.
- Understand the features, pros/cons of languages.
- Do things in the right way.
- Learn a new language effectively and quickly.
- Choose a language to use for a project/system.
- Design new languages.


Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 20



CSE 3302: What isn't it about?

- I want to get easy credit without doing much.
- I want to become a coding guru in C/C++/C#/Java/...
- I want to know every details about implementing a language.
- I want to prove properties of a language.


Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 21



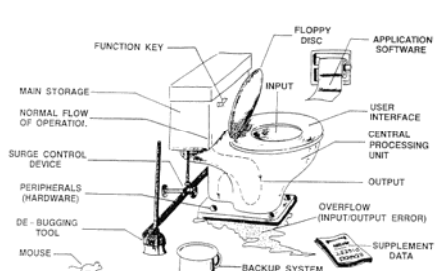
Big picture

- Let's draw the picture

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 22




Big picture?

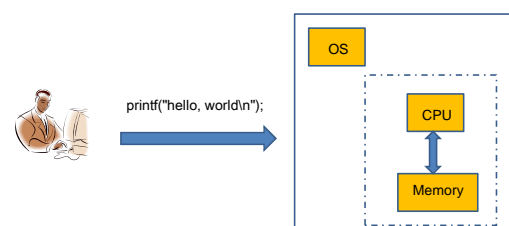


<http://thunder.prohosting.com/~ccastle/MyComputer.gif>

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 23



What is a programming language?




printf("hello, world\n");

- Programmers use PLs to instruct computers to do things.

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington
©Chengkai Li, 2007 24

Programming languages?



LDA #2
STA X

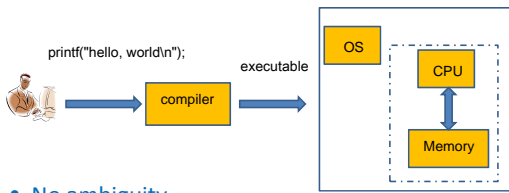
SELECT name
FROM Students
Where status="sleeping"

(http://encarta.msn.com/media_461557622/ENIAC.html)

- We talk about high-level general-purpose programming languages.

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 25

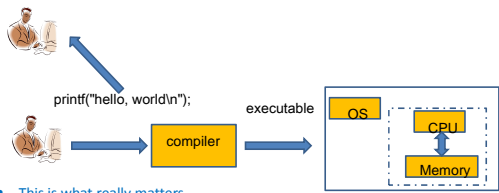
Machine Readability



- No ambiguity
- Efficient translation/compilation
- ...
- Covered in compiler course

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 26

Human Readability



- This is what really matters
 - Abstractions for describing the actions
 - Data abstraction: subject of computation
 - Control abstraction: transfer of control in computation
 - Human-Human communications
 - Your program will be read by others

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 27

Paradigms

- Imperative/Procedural:** (ALGOL, FORTRAN, Pascal, C, Ada)
control statements manipulate the data and program states
- OO:** (C++, Java, Smalltalk)
data/object-centric
- Functional:** (Lisp, Scheme, ML, Haskell)
everything is done by function evaluations
- Logic:** (Prolog)
axioms (statements about truth) + inferences
- Parallel:** (threads in Java)
- Mix:** (Python, Perl, Ruby)

Few "pure" languages

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 28

Syntax and Semantics

- Syntax:**
structure of programs (tokens, keywords, statements.)
- Semantics:**
What are meant by programs

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 29

Schedule:

What we are going to learn

• Introduction	(1.5 lectures)
• History	(0.5 lectures)
• Syntax	(2 lectures)
• Semantics	(2 lectures)
• Data Types	(2 lectures)
• Control	(3 lectures)
• Abstract Data Types	(2 lectures)
• OO (C++, Java, Smalltalk)	(3 lectures)
• Functional (Lisp, Scheme, ML, Haskell)	(3 lectures)
• Logic (Prolog)	(2 lectures+1 guest lecture)
• Python, PHP, Perl, Ruby	(2 lectures: student discussion)
• A taste of advanced topics and research	(2 guest lectures)

Lecture 1 - Introduction, Fall 2007 CSE3302 Programming Languages, UT-Arlington ©Chengkai Li, 2007 30