CSE 3302
Programming Languages

Chengkai Li

Spring 2008

Self Introduction

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

Now it's your turn…

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

Course Page

- http://crystal.uta.edu/~cli/cse3302
  Syllabus, Announcements, Schedule (lecture notes), Homework, Machine Problems, Exams, Resources, Accommodation based on disability.
- Important announcements:
  announcement page (http://crystal.uta.edu/~cli/cse3302/announcements.htm)
- Tentative!

Basics

- **Lectures:** Tuesday / Thursday , 3:30-4:50pm, GACB 105
  - Slides don't cover everything. You need to read the book and supplementary materials.
- **Instructor: Chengkai Li**
  Office hours: Fri. 10am-12pm , Nedderman Hall 334
  Contact: (817) 272-0162, cli [at] uta.edu
- **Assistant Instructor/TA: Weimin He**
  Office hours: Tu/Th 5-6pm, GACB107
  Contact: (817) 272-7618, wmhe [at] exchange.uta.edu
- **Textbook:** *Programming Languages - Principles and Practices*, 2nd Edition, by Kenneth C. Louden.
- **Reference book:** *Principles of programming languages: design, evaluation, and implementation,* 2nd Edition, by Bruce J. MacLennan.

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): 15%, individual.
- Machine Problems (MP): 30%, individual.
- Essays (ES): 10%, individual.
- Midterm exam: 20%,
- Final exam: 25% ,
- Bonus points: 5%, based on class participation.

Homework (HW) 15%

- HW1, HW2, HW3, HW4
- Problem-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, MP4.

- Can discuss, but must do it independently.

- Decent amount of programming:
  - Not crazy.
  - But not a piece of cake either.

Essays (ES) 10%

- satisfying ABET requirements
- Can discuss, but must write it independently.

- Must get at least 37.5 (out of 100) on essay, otherwise you will get Incomplete (I) as your grade, no matter what your scores on other components are.

- panel discussion based on essay.

Deadlines

- 11:59pm.
- After due time, deduct 5 points per hour, till you get 0.

(No exception, unless you have medical or other emergency, with proof documents from the school.)

WebCT

- Assignment instruction and files
- Student assignment submission (we don't accept email submission or hard-copy)
- Discussion Group
- Grades

Exam

- Midterm: 20%
  Feb. 28th Review
  Mar. 04th Exam (in class), close-book, close-notes
- Final: 25%
  May. 1st Review
  May. 08th Exam, Thursday, 2-4:30pm, close-book, close-notes

Regrade

- 7 days after we post scores on WebCT and send out announcements. 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:
  - More details later

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=gC2ew6qLa8U

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

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

read & sign the statement

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.

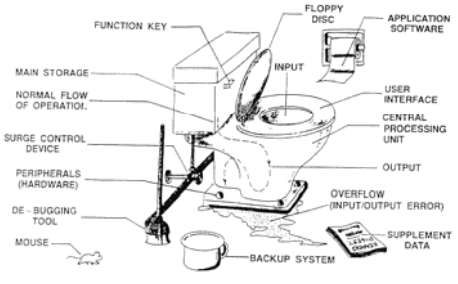
CSE 3302: What isn't it about?

- 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.

Big picture

- Let's draw the picture

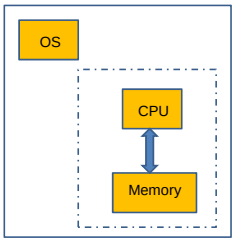
Big picture?



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

What is a programming language?

printf("hello, world\n");



- Programmers use PLs to instruct computers to do things.

Programming languages?

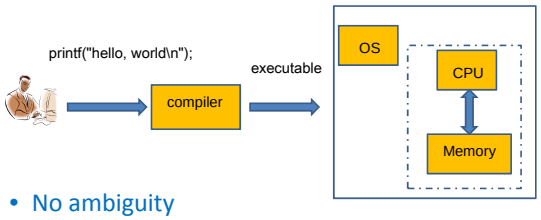
(http://encarta.msn.com/media_461557622/ENIAC.html)

```
LDA  #2
STA  X
```

```
SELECT name
FROM   Students
Where  status="sleeping"
```

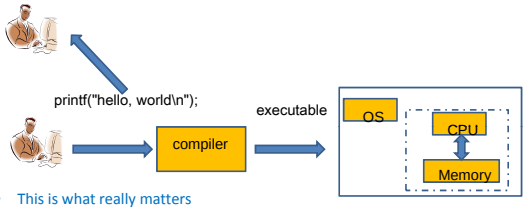
- We talk about high-level general-purpose programming languages.

Machine Readability



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

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

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

Syntax and Semantics

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

Schedule: What we are going to learn

- Introduction (1 lectures)
- History (0.5 lectures)
- Syntax (2 lectures)
- Semantics (2 lectures)
- Data Types (1 lectures)
- Control (2 lectures)
- Abstract Data Types (1 lectures)

- OO (Java, Smalltalk) (6 lectures)
- Functional (Lisp, Scheme, ML, Haskell) (4 lectures)
- Logic (Prolog) (4 lectures)
- Python (2 lectures: panel discussion)
- A taste of advanced topics and research (2 guest lectures)