# Illinois State University

**ITK 327/327.05, Fall 2006**

# Concepts of Programming Languages

**Old Union 213E, MW 11:00 ~ 12:15 PM**

Instructor: Chung-Chih Li, Ph.D.
http://www.itk.ilstu.edu/faculty/chungli
cli2@ilstu.edu
Office: Old Union 105, Tel:(309) 438-7952
Office Hours: MTWT 2:30 ~ 3:30 PM or by appointment

**Classroom and meeting time: (Attendance will be taken impulsively)**

**Old Union 213E, MW 11:00 ~ 12:15 PM**

**WebPage of the course:** http://www.itk.ilstu.edu/faculty/chungli/ITK327

Students should check the Webpage of the class from time to time. From there, you may find important information about assignments, data, due dates, sample programs, or announcements.

**Note:** *An announcement made in the class will be considered as an official one, since I may not be able to update every announcement.*

**Prerequisites:**

ITK 179 with a C or better grade.

We assume that students already have fair experience and skill to develop programs in at least one contemporary programming language.

**Textbooks:**

*Modern Programming Languages: A Practical Introduction,* by Adam Brooks Webber, Publisher: Franklin, Beedle & Associates, 2003.

**Course Description and Purposes:**

In this course, we will study the most important parts that form the modern programming langues and the most important concerns that need to be taken care of. We will learn how is the syntax of a programming language to be specified? Why should we be specific on declare *types*? Can we not? Why should we care about the scopes? How do we handle the parameters? These are just some examples around the center problems of modern programming langues. We will kick around these problems in this course. However, we are not going to study "how" to actually implement those concepts we just mentioned (that belongs to Compiler Construction course); instead, we study what are those concepts and why we need them from the aspect of both the user and the implementer.

Although most of the programming languages are designed to be general, they can be further characterized into a four major categories: imperative language, functional languages, logical language, and objected-oriented languages. We will compare the nature of each category and study two cases: ML (a functional language) and Prolog (a logical language).

**Examinations:** (440 points) Two midterms and one final exam; 120 points for each midterm and 200 points for the final.

- Unless announced otherwise, all tests are accumulative, closed book, and indispensable. No makeup test will be given unless a documented absence is authorized by the university.

- Every student is allowed to bring a *self-prepared hand-writing* crib sheet to the test. You can **write** down anything on both sides of **one** letter-sized paper. No circulation during the test.

| | | | |
|---|---|---|---|
| $6^{th}$ week's | Midterm I | 120 points | Sep. 25, Monday |
| $12^{th}$ week's | Midterm II | 120 points | Nov. 6, Monday |
| $17^{th}$ week's | Final Exam | 200 points | Dec. 13, Wednesday, 7:50 AM |

**Assignments:** (500 points) About 11 to 12 assignments will be given, and three of them will be programming assignments. The weight of each assignment depends on the difficulty of the assignment; usually between 20 and 40 points for non-programming assignments and between 40 and 60 for programming assignments.

**Read the following requirements carefully; they are applied to all assignments!**

• For non-programming assignment, you should prepare your works according to the following guideline for submission.

1. A cover page with your name and student ID on it.
2. Your solutions; handwriting is acceptable *only if* your writing is clear and with appropriate spacing and organization.
3. Staple all pages together and put them in a letter-sized Manila folder with your name on it. If your paper is torn from a spiral notebook, fine, but trim the edge.

• For programming assignment, you should prepare your works according to the following guideline for submission.

1. Put a few comment lines at the beginning of your program file, in which you should clearly indicate your name and ID and claim your copyright. **Student who fails to do so will receive 0 point on the assignment.**
2. Submit your assignment with items in the following order:
   (a) A cover page with your name and student ID on it.
   (b) A brief summary about the assignment and your approach to the problem. You may include the difficulties you had faced, if any, or why you think your program doesn't work. It is very common and not a shame to admit that your program doesn't work under time constraints, but a reasonable self-diagnosis deserves reasonable partial credit.
   (c) A hard-copy of the source codes.
   (d) A hard-copy of the directly output of your program, if any.
   (e) A diskette, R/W CD, or USB memory stick containing all source and byte codes *under the root directory*. Don't use DVD.
   (f) All item above should be put in a letter-sized Manila folder with your name on it.

   All materials will be returned after graded, and you are encouraged to reuse them, if appropriate. However, you should prepare at least two sets, because I will not be able return your work before your next assignment begins.

Do backup your works as often as possible. Remember: bad things do happen, and "my dog ate my works" is not a good excuse.

Students are encouraged to discuss assignments and help each other. However, this does not mean that you can either entirely or partially copy or modify other's works.

**Try very hard to avoid the following troubles:**

1. Any form and any degree of plagiarism will receive 0 point.
2. If your program contains syntax error, i.e., I can't compile the program, you will receive 0 point. Note well that the difference between "my program can't be compiled" and "my program doesn't work" is huge. The former one deserves nothing.
3. If the hard-copy of the direct output of your program is inconsistent to your program's design, you will receive 0 point. This is a kind of cheating.
4. Late works will be graded with penalty: -10 points per day after the due date. A weekend is counted as 2 days.

**Attendance:** Attendances will be taken impulsively. Each unauthorized absence will cost you 50 points from your score tally.

**Pop quizzes:** ( 100+ points)

Some pop quizzes will be given without notice in advance. Each quiz carries 20 points towards students' final scores. The coverage of every quiz is also accumulative, including the materials that are three-month-old and those covered in the class right before the quiz. A typical quiz takes about 10 minutes. No makeup quiz will be given if missed. If you miss a quiz due to a university authorized absence, we will use the average of your rest quizzes as the score; otherwise, you get a 0 for the missing quiz.

**Academic Honesty:**

Cheating, plagiarism, collusion, abuse of resource materials, and their consequences are defined and described in ISU 2006-2007 Undergraduate Catalog, Section: Academic Policies and Practices, Article: Academic Integrity (Page 63) and Code of Student Conducts under `X.C.` Disciplinary Bodies And Procedures – Academic Honesty Cases .

Students giving away academic works for assignment offered for credit to other students working on the same assignment will be considered as guilty as academic dishonesty, and will receive the same penalty.

More information can be found at:

`http://www.deanofstudents.ilstu.edu/crr/downloads/Code_of_Student_Conduct.pdf`

**Grading Policy:**

You may have more than 1000 points to gain; but we always consider 1000 points as the perfect score. your grade is based on the scheme shown in the following table.

| Points | Grade | |
|---|---|---|
| 850 $\sim$ up | A | Excellent |
| 750 $\sim$ 849 | B | Good |
| 600 $\sim$ 749 | C | Satisfactory |
| 500 $\sim$ 599 | D | Passing |
| 0 $\sim$ 499 | F | Failure |

## I do not curve!!

I'm not afraid to give all A's, neither am I to give all F's. In other words, you don't have to knock down your friends to get a good grade, so do help them if they need you, and you can't hide behind someone else, because you two could be both shot down.

**Tentative Topics and Schedule:**

Keep the table of tentative topics and schedule in the following handy, and try to keep up with the schedule. Read the assigned materials before the class.

| Week | Topics | Reading |
|---|---|---|
| 0: – | Kicking off | Syllabus |
| 1: Aug. 21 | Characterization of Programming Languages | Chapter 1 |
| 2: Aug. 28 | Grammar and Syntax of Programming Languages | Chapters 2 |
| 3: Sep. 4 | Parsing, Syntax v.s. Semantics | Chapter 3 |
| 4: Sep. 11 | Language System – how they work? | Chapter 4 |
| 5: Sep. 18 | Types | Chapter 6 |
| 6: Sep. 25 | **(Midterm 1)** Polymorphism | Chapter 8 |
| 7: Oct. 2 | More on Polymorphism, Scopes | Chapters 8, 10 |
| 8: Oct. 9 | Memory Management | Chapter 14 |
| 9: Oct. 16 | Parameters | Chapter 18 |
| 10: Oct. 23 | ML – A functional programming language. | Chapters 5, 7 |
| 11: Oct. 30 | More on ML | Chapter 9, 11 |
| 12: Nov. 6 | **(Midterm 2)** Prolog – A logical programming language | Chapters 19, 20 |
| 13: Nov. 13 | More on Prolog | Chapters 22 |
| 14: Nov. 20 | Thanksgiving break | review/catch up |
| 15: Nov. 27 | Object Orientation | Chapter 16 |
| 16: Dec. 4 | Java | Chapters 13, 15, 17 |
| 17: Dec. 11 | Final Examination, Dec. 13, Wednesday, 7:50 AM | all |

## Play Ball!!