CSc 520

Principles of Programming Languages

0: Administrivia

Christian Collberg

collberg@cs.arizona.edu

Department of Computer Science

University of Arizona

Contact Information

Class	:	520 PRINCIPLES OF PROGRAMMING LANGUAGES
Lecturer	:	Christian Collberg
Email	:	collberg@cs.arizona.edu
WWW	:	http://www.cs.arizona.edu/∼collberg
Office	:	758
Office Hours	:	Open door policy
Phone	:	621-6612
Lectures	:	3:00–4:15, MW, GLD-S 701
Book	:	Programming Language Pragmatics. Michael Scott
TA	:	TBA

Course Outline (Subject to change)

- This course will defi ne, analyze and evaluate important concepts found in current programming languages.
- Its goals are to build an ability to evaluate and compare programming languages.
- We will evaluate and compare languages both from the user's and implementor's view.
- We will develop precise mechanisms for specifying the semantics of programming languages.

Course Outline (Subject to change)...

In particular, we will cover the following topics:

1. scope of objects and time of binding
2. module mechanisms (e.g., blocks, procedures, coroutines)
3. data abstraction, datatypes
4. control structures
5. storage management and runtime support
6. operational, denotational, and axiomatic semantic specifi cation; attribute grammars
7. applicative and object-oriented languages

Grading (Subject to change)

1. One final exam (50%), Wednesday, May 11, 14:00–16:00.
   (a) The exam is closed book.
   (b) Without prior arrangement, missed exam $\Rightarrow$ grade of zero.
   (c) Fail the exam $\Rightarrow$ you might fail the course.
2. "Several" homework assignments (50%) . Homeworks may require programming, theoretical work, or paper presentations.

Grading (Subject to change)...

If your graded score for an homework is g and you handed in k days late, then your computed score for this assignment will be

$$\begin{cases} \max(0, g(1-0.1k)) & \text{if } k \leq 5 \\ 0 & \text{otherwise} \end{cases}$$

Grading...

- I reserve the right to scale final grades.
- Scaling can be both "up" and "down", depending, for example, on whether the final exam turned out to be particularly easy or hard.

You cannot make up an in-class test unless

1. you have notified us in writing (email is fine) or by phone prior to the test that you will be absent, and
2. you can produce a note from your doctor saying that you were incapable to take the test, and
3. you receive permission from the instructor prior to the test.

Attendance Policy

- You are not required to attend lectures, but...

  you cut class at your own risk.

  Anything covered in class or in any of the required readings is fair game on tests and exams.

Subject to Change Policy

- We may add, drop, or change topics.
- We may change exam or homework dates, etc.
- Changes will be announced in class and on the class web site!
  You are responsible for checking this site regularly.
- You should also check the course newsgroup `cs.course520` for announcements.

Notification of Objectionable Materials

- Some of the material may be hard, boring, or both.
- The instructor is known to sometimes make jokes in class which
  a) are not funny, or
  b) may be slightly off-color.

  He apologizes in advance.
- Assignments and examples may touch on subjects which some may deem questionable, such as sex, war, and rock'n'roll.

Handicapped Accessibility

Students with disabilities who require reasonable accommodations to fully participate in course activities or meet course requirements must register with the Disability Resource Center. If you qualify for services through DRC, bring your letter of accommodations to me as soon as possible. See `http://drc.arizona.edu`.

Student Code of Academic Integrity

- Assignments in this course require individual attention and effort to be of any benefi t. All work is expected to be that of each student alone. You may not consult with others, except in ways specifi cally authorized by the course instructor. You also may not plagiarize another person's work or copy another person's code.

Student Code of Academic Integrity...

- Code of Academic Integrity. Students are responsible for understanding and complying with the University's Code of Academic Integrity. A synopsis of the Code is attached; the full text is available from the Offi ce of the Dean of Students in Room 203 Old Main. Among other provisions, the Code demands that the work you submit is your own, and that graded papers and exams will not subsequently be tampered with. Copying of another student's programs or data, or writings is prohibited when they are part of a published class assignment; it is immaterial whether the copying is by computer, xerox, pen or other means. Witting collaboration in allowing such copying is also a Code violation.

Student Code of Academic Integrity...

- Assignments in this course require individual attention and effort
- Violations of the Code will, at minimum, result in loss of credit for a graded item. An egregious fi rst violation or any second violation will minimally result in failure of the entire course.
- See also
  `http://info-center.ccit.arizona.edu/~studpubs/policies/cacaint.htm`
  the University of Arizona Code of Academic Integrity.

I take academic integrity seriously! I will report *every* violation!

Course Methodology

- I will lecture, you will learn.
- You will participate in class discussions.
- It is important in this class to allot signifi cant time outside of class to programming in the new languages we will study.
- You cannot pass this class by cramming before the fi nal.
- Most of the languages we will study have free implementations. If you own your own computer it's a good idea to download and install the interpreters so that you can work at home.

Prerequisites, Required Knowledge

- Prerequisites: C Sc 453, or equivalent background in Compilers.
- You need to be a competent programmer in a procedural/object-oriented language, such as Java or C.

Learning Languages

- As part of this class we will study a few languages in detail.
- While I will lecture on each language, you will be responsible for reading up on the details of the languages. This will involve getting out books from the library, searching the web for relevant tutorials, etc.
- There may be small programming assignments for these languages.

Handouts & Other Material

1. I always make copies of my transparencies available to students. Note that
   - I do this to relieve you of having to take notes during lectures,
   - they are not substitutes for reading the textbook,
   - their primary purpose is to remind you of what you need to study for the exam.

Handouts & Other Material...

2. Various manuals and papers will be handed out during class. Extra copies can be picked up from the boxes outside my offi ce.
3. Various information regarding the course (including postscript fi les of the handouts) can be found on the info-bahn:
   `http://www.cs.arizona.edu/~collberg/Teaching/520/2005/index.html`

Free Compilers and Interpreters

Ada:
`ftp://wuarchive.wustl.edu/languages/ada/compiler/gnat/distrib/3.14p`.

Scheme: `http://www.drscheme.org`.

Modula-2:
`http://floppsie.comp.glam.ac.uk/Glamorgan/gaius/web/GNUModula2.html`.

Modula-3: `http://www.m3.org` or `http://www.elegosoft.com`.

Gofer: `ftp://ftp.dcs.gla.ac.uk/pub/haskell/gofer`.

Also see `http://www.idiom.com/free-compilers`.

Installed Translators

- Compilers and interpreters available on lectura :

  Pascal: `gpc`

  Scheme: `scheme` and `scheme48`.

  C,C++,Objective-C: `gcc`.

  ML: `sml`.

  Icon: `icont`.

  Prolog: `prolog`.

- Compilers and interpreters available on linux :

  Gofer: `/home/cs520/2003/bin/linux/gofer`

Now What?

Let's Have Fun!!![a]

[a] That's right —learning about programming languages is fun!