CSc 520 — Principles of Programming Languages

0 : Administrivia

Christian Collberg
Department of Computer Science
University of Arizona
`collberg+520@gmail.com`

January 16, 2008

1 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	:	15:00–16:15, MW, GLD-S 701
Book	:	Programming Language Pragmatics. Michael Scott
TA	:	TBA

2 Course Outline (Subject to change)

- This course will define, 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.

3 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 specification; attribute grammars
7. applicative and object-oriented languages

4 Grading (Subject to change)

1. One final exam (50%) .
   (a) The exam is closed book.
   (b) Without prior arrangement, missed exam ⇒ grade of zero.
   (c) Fail the exam ⇒ you will fail the course.
2. "Several" homework assignments (50%) . Homeworks may require programming, theoretical work, or paper presentations.

5 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.

6 Syllabus

You are responsible for reading
and understanding this syllabus.
If you have any concerns or issues
about the information in this document
you should bring them up during the
first week of class.

7 Required extracurricular activities

- Programming assignments.
- Homework assignments.
- Reading the textbook.
- Working programming exercises on your own.

8 Special materials required for the class

- None.

9 Assignment Format

- Assignments will be mostly in the form of programming problems.
- You may work the assignments on any machine you want, but before you hand them in *you should test the code on lectura!* I will grade the assignments on lectura, and if they don't work there, I won't debug them for you!
- There can be subtle problems with code that's developed on a Windows machine, for example, when it is run on a Unix machine. For example, the two systems use different newline characters.

Assessment Scheme

10 Tests, Quizzes, and Assignments

There will be

1. one comprehensive final exam, worth a total of 50%;
2. assignments, worth a total of 50%.

11 Late Assignments

- Assignments handed in no more than 24 hours late will incur a 10% penalty.
- Assignments handed in more than 24 but no more than 48 hours late will incur a 20% penalty.
- Assignments handed more than 48 hours after the deadline will receive a grade of 0.

12 Making up Tests

You cannot make up the midterm or final exam unless

1. you have notified the instructor in writing (email is fine) or by phone prior to the test that you will be absent, and
2. you receive permission from the instructor to take the test at a later date.

13 Curving

- All grades (for exams, quizzes, and assignments) will be curved up by throwing away the highest grade in the class and scaling up such that the second highest grade is 100.
- The curving is done to adjust for particularly difficult tests/assignments, and to prevent an outlier from skewing the grade distribution.
- You cannot, after scaling, receive more than 100 on any exam, quiz, or assignment.

14 Grade Assignment

- You will fail the class if you get less than 50 (after curving) on the final exam.
- Otherwise, a curved total grade of [90,100] gives you an A, [80,89] a B, [70,79] a C, [60,69] a D, and 59 and below an E.

15 Incomplete work policy

- Except under exceptional circumstances I will not assign incomplete grades.
- I decide what is an exceptional circumstance.

16 Detailed Grading Scheme

- To avoid any ambiguities, I have formalized the informal rules given above.
- The rules below should be considered *minimum* requirements to achieve a particular grade. The instructor reserves the right to do additional adjustments, as necessary.
- Any contradictions, omissions, errors, or ambiguities in the grading scheme will be resolved by the instructor.
- Any issues or concerns regarding the grading scheme should be brought to the attention of the instructor within the first week of class.

17 Details — Curving

- All raw scores range from 0 to 100.
- Each individual score (final, midterm, quizzes, assignments) will be curved using the function

  $$\mathrm{curve}(x, s) = \min(100, (100.0/\max(x - \max(x)))x_s)$$

  where x is a set of scores (for an assignment, a test, etc.) and s is a student.
- Note: $-$ is set subtraction.
- $\mathrm{curve}(x, s)$ returns s's score, curved up by 100.0/*2nd_highest_class_score*.

18 Details — Curving...

- For example, assume the following final exam scores:

  34 45 66 88 98

  After the curve has been applied, the scores will be

  38.6 51.1 75 100 100

19 Details — Exams

final exam:

- Let f be the set of final exam scores.
- Let f^s be the final exam score for student s.
- Let $\mathcal{W}^f$ be the weight of the final exam (50%).
- $t^s_f = \text{curve}(f, s)\mathcal{W}^f$ is the curved final score for s.

20 Details — Assignments

- Let a_i be the set of scores for the i:th assignment.
- Let a^s_i be the score for student s on the i:th assignment.
- Let $\mathcal{W}^a_i$ be the weight of the i:th assignment ($\sum_i \mathcal{W}^a_i = 50\%$).
- Let α^s_i be the assignment score after late penalties have been applied:

$$\alpha^s_i = \begin{cases} a^s_i & \text{if the assignment is handed in on time} \\ 0.9a^s_i & \text{if the assignment is} > 0 \text{ and} \leq 24 \text{ hours late} \\ 0.8a^s_i & \text{if the assignment is} > 24 \text{ and} \leq 48 \text{ hours late} \\ 0 & \text{if the assignment is} > 48 \text{ hours late} \end{cases}$$

21 Details — Assignments. . .

- $t^s_a = \sum_i (\text{curve}(\alpha_i, s)\mathcal{W}^a_i)$ is the total curved assignment score for student s.
- If, for whatever reason, the actual number of assignments is less than the planned number, the $\mathcal{W}^a_i$'s will be scaled up uniformly.

22 Details — Total Scores

- The raw total score for student s is

$$t_s = t^s_f + t^s_a$$

- We round up to the nearest integer:

$$\text{total}_s = \lceil t_s \rceil$$

23 Details — Grade Assignment

- The final grade assignment for student s is

$$\text{grade}_s = \begin{cases} E & \text{if } t^s_f < 50 \\ \begin{cases} A & \text{if total}_s \in [90, 100] \\ B & \text{if total}_s \in [80, 89] \\ C & \text{if total}_s \in [70, 79] \\ D & \text{if total}_s \in [60, 69] \\ E & \text{if total}_s < 60 \end{cases} & \text{otherwise} \end{cases}$$

- In other words, a student with a curved final exam score $t^s_f < 50$ will fail the class, regardless of their results on the other assessment categories.

Policies

24 Office hours

- I use an open door policy: