CS 354: Programming Languages

Instructor

Instructor:	Jim Buffenbarger
Office:	CCP-359
Voicemail:	(208) 426-3567
Email:	mailto://buff@cs.boisestate.edu
WWW:	http://csweb.boisestate.edu/~buff

Meetings

Lectures:	TuTh 1:30–2:45	CCP-243
Office hours:	TuTh 2:45–3:45	CCP-359
	by appointment	CCP-359

Lectures will be audio/video recorded, and made available, afterwards.

Our Teaching Assistant is:

    TBA

TA office hours are at (this "**pub**" directory is explained, below):

pub/TutorOfficeHours

I am happy to answer questions by email. Please please click here, or see:

pub/doc/EmailQuestions.pdf

Catalog Description

Principles of programming languages: design, syntax, semantics, information binding, strings, arithmetic, input/output, recursion and extensibility.

PREREQ: CS 321.

In addition, familiarity with Unix, C, and Java is assumed.

Goals

At the end of the course, students will be able to do the following:

- identify characteristics of procedural, object-oriented, functional, and scripting languages

- describe the phases of program translation

- explain different forms of binding, visibility, scoping, and lifetime management

- demonstrate the differences between various parameter-passing methods

- explain the concepts of encapsulation, abstraction, inheritance, and polymorphism

- write programs in languages based on several different programming paradigms

- evaluate a language on the basis of the various features which it supports

Students also experience working on a team, developing a website, and giving an oral presentation.

Textbook

Programming Language Pragmatics, by Michael L. Scott, Fourth edition, 2015, Elsevier: Morgan Kaufmann.

Other Course Material

This syllabus, lecture slides, assignments, and other material is available in what we'll called our "pub" directory, on our Computer Science Lab (CCP-240, CCP-241, and CCP-242) computers, served by onyx.boisestate.edu. The directory is at:

```
~jbuffenb/classes/354/pub
```

This directory is read-only. So, you might want to copy it, perhaps to your local computer. Since onyx services Secure Shell (SSH) requests, you can use SSH clients (e.g., scp and sftp) to do so. However, beware: It contains symbolic

links to parent directories, and `scp -r` will unconditionally follow them, thereby looping forever. To avoid this, use `sftp` or `tar`/`ssh`, as needed.

A copy of this directory can be accessed from our Canvas website, via the "Files" tab on the left sidebar. This copy is updated less frequently, since bulk upload is slow. Furthermore, although lecture slides can be viewed, the Canvas PDF viewer will not follow links to other files.

A copy of this directory can also be accessed via GitHub.

You may also find the following local guide useful:

> `pub/etc/cs-linux.pdf`

In particular, it explains how to use SSH.

Grades

At the end of the semester, the Registrar requires a letter grade for each student. Accordingly, during the semester, homework and exams are evaluated, and numeric scores are assigned. Each such artifact is worth a certain number of points, and has a weight. From these scores, a student's overall numeric/raw percentage is computed.

Collectively, a class's raw percentages form a distribution, with a mean and standard deviation, sometimes called a Bell Curve. An algorithm normalizes these values into a "grading" distribution, with a desired mean and standard deviation of 85 and 10 (i.e., the values are "curved"). A normalized value is then mapped to a letter grade, in the conventional way. If you are interested in the gory details, click here.

During the semester all of this evolving data can be found in our `pub` directory, on `onyx`, its nodes, Canvas, and GitHub.

Homework is due at 11:59PM, Mountain Time, on the day it is due. Late work is not accepted. To submit your solution to an assignment, login to a lab computer, change to the directory containing the files you want to submit, and execute:

> `submit jbuffenb` *class assignment*

For example:

> `submit jbuffenb cs354 hw1`

The `submit` program has a nice `man` page.

When you submit a program, include: the source code, sample input data, and its corresponding results.

Scores are posted in our `pub/scores` directory, as they become available. You will receive a code, by email, indicating your row in the score sheet. You are encouraged to check your scores to ensure they are recorded properly. If you feel that a grading mistake has been made, contact me as soon as possible.

The weights of homework and exams is shown in the table below:

Activity	*Weight*
Homework Assignments	40%
Translator Assignment	20%
Language Website	20%
Exam	10%
Final	10%

Homework Assignments (HW)

Students are assigned several problem sets, composed of the following:

- Exercises are selected from the end of each chapter of the textbook.

- Small programming problems are assigned, to be implemented in what are expected to be unfamiliar programming languages (e.g., Scheme). Translators for these languages are available on the Linux computers in the Computer Science lab.

Students work on these assignments individually, not as teams.

The rubric that will be used to grade each assignment is distributed with the assignment. Try to focus on the assignments, rather than the rubrics.

Translator Assignment (TA)

Students extend a Java implementation of a simple programming-language translator (i.e., a combination interpreter/compiler). A Java development environment is available on the Linux computers in the Computer Science lab.

Students work on this assignment individually, not as teams.

The rubric that will be used to grade this assignment is distributed with the assignment.

Language Website (LW)

Students are randomly partitioned into teams, of 3–5 members. Each team chooses an unfamiliar programming language, and develops a website dedicated to that language. Website content includes example programs of varying complexity. During the last couple of weeks of the semester's lectures, each team introduces their language, and describes the experience of programming in that language, in an in-person team-delivered oral presentation.

Of course, team members work together on this assignment.

The rubric that will be used to grade this assignment is distributed with the assignment, but scores will not be made available before the end of the semester.

Exams

An exam and a final are administered. These are in-class, open-note, open-textbook, and open-laptop tests. Students work on these individually.

Attendance

In-person lecture attendance is an important part of course participation. Attendance is taken at each lecture: starting five minutes before the scheduled start time, and ending fifteen minutes after the scheduled start time. Attendance is not taken during the first week of classes, holidays, or finals week.

Attendance can affect your grade. Each absence results in a one-percent reduction of your overall normalized percentage. Since a few absences are expected, completion of BSU's on-line end-of-semester course evaluation will erase up to five absences.

Attendance is administered wirelessly, via the iClicker app, available for free, from your smartphone's app store. For more information, click here.

Source-Code Documentation

Good documentation and programming style is very important. Your programs must demonstrate these qualities for full credit. Good documentation and programming style includes:

- heading comments giving: author, date, class, and description

- function/procedure comments giving description of: purpose, parameters, and return value

- other comments where clarification of source code is needed

- proper and consistent indentation

- proper structure and modularity

For more information, and examples, click here.

Academic Integrity

The University's goal is to foster an intellectual atmosphere that produces educated, literate people. Because cheating and plagiarism are at odds with that goal, those actions shall not be tolerated in any form. Academic dishonesty includes assisting a student to cheat, plagiarize, or commit any act of academic dishonesty. Plagiarism occurs when a person tries to represent another person's work as his or her own or borrows directly from another person's work without proper documentation.

If a student engages in academic dishonesty, the student may be dismissed from the class and may receive a failing grade. Other penalties may include suspension or expulsion from the University.

For much more information about academic integrity, including examples of academic dishonesty, please click here. If you are unsure about a particular behavior, ask your instructor.

Labs and Safety

Each student receives an account on the cluster of computers in the Computer Science Labs: CCP-240, CCP-241, and CCP-242. The cluster comprises a server named onyx.boisestate.edu and a set of nodes with shared home directories. It is remotely accessible, via SSH. The cluster runs the Linux operating system.

Physical access requires building and room access. After-hours building access, and all-hours room access, require an authenticated proximity-type student-identification card.

You are responsible for understanding and obeying lab rules.

Schedule

Week	Date	Topic	Assigned	Due	Reading
1	Aug 22 Tue	Introduction			1
	Aug 24 Thu		HW1		
2	Aug 29 Tue	Programming Language Syntax			2.0-2.1
	Aug 31 Thu			HW1	
3	Sep 05 Tue		HW2		
	Sep 07 Thu				
4	Sep 12 Tue				
	Sep 14 Thu			HW2	
5	Sep 19 Tue	Names, Scopes, and Bindings	TA		
	Sep 21 Thu				
6	Sep 26 Tue				
	Sep 28 Thu				4.0-4.1
7	Oct 03 Tue				
	Oct 05 Thu	Control Flow			6
8	Oct 10 Tue	Exam			
	Oct 12 Thu			TA	
9	Oct 17 Tue		HW3		
	Oct 19 Thu	Data Types			7-8
10	Oct 24 Tue		LW		
	Oct 26 Thu				
11	Oct 31 Tue	Subroutines and Control Abstractions			9.0-9.4
	Nov 02 Thu				
12	Nov 07 Tue			HW3	
	Nov 09 Thu		HW4		
13	Nov 14 Tue				
	Nov 16 Thu		HW5	HW4	
14	Nov 21 Tue	Thanksgiving			
	Nov 23 Thu	Thanksgiving			
15	Nov 28 Tue	LW Presentations		LW	
	Nov 30 Thu	LW Presentations		HW5	
16	Dec 05 Tue	LW Presentations			
	Dec 07 Thu	LW Presentations			
17	Dec 12 Tue	Final: 2:30-4:30			