

CS 354: Programming Languages

Instructor

Instructor: Jim Buffenbarger
Office: CCP-359
Email: buff@cs.boisestate.edu
Phone: 208-426-3567
WWW: <http://cswb.boisestate.edu/~buff>

Meetings

Lectures: MoWe 1:30–2:45 CCP-221
Office hours: TuTh 2:00–3:00 CCP-359
by appointment CCP-359

For BSU COVID-19 Information, please see:

<http://cswb.boisestate.edu/~buff/files/covid.pdf>

Our Teaching Assistants are Shefali Keshri and Mushfiqur Rahman:

shefalikeshri@u.boisestate.edu
sheikhmdmushfiqur@u.boisestate.edu

I am happy to answer questions by email. Please see:

<http://cswb.boisestate.edu/~buff/files/EmailQuestions.pdf>
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, the student will be able to 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*, Michael L. Scott, Fourth edition, Elsevier: Morgan Kaufmann, 2015, ISBN: 9780124104099.

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.

On our Canvas website, a copy of this directory can be accessed 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.

Office hours for our Teaching Assistants are at:

```
onyx:~jbuffenb/classes/354/pub/TutorOfficeHours
```

You may also find the following local guide useful:

```
onyx:~jbuffenb/classes/354/pub/etc/cs-linux.pdf
```

In particular, it explains how to use SSH.

Links to programming-language documentation can be found at:

```
http://cswb.boisestate.edu/~buff/pl.html
```

Grading

At the end of the course, a letter grade is assigned to each student according to rank among classmates, which is determined from numerical scores assigned for performance of these activities:

<i>Activity</i>	<i>Weight</i>
Textbook Assignments	10%
Language Assignments	30%
Interpreter Assignments	20%
Language Website	20%
Exam	10%
Final	10%

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 cs101 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.

Textbook Assignments (TA)

Several problem sets are assigned, from the exercises at the end of each chapter of the textbook. Students work on these individually, not as teams.

Language Assignments (LA)

Several programs are assigned, to be developed 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 individually, not as teams.

Interpreter Assignments (IA)

A couple of programs are assigned, to extend a provided Java implementation of a simple programming-language interpreter. A Java development environment is available on the Linux computers in the Computer Science lab. Students work on these individually, not as teams.

Language Website (LW)

Each team of students develops a website dedicated to a particular, unfamiliar, programming language. Teams are formed, and languages are assigned, randomly. Several milestones are assigned. Open-source translators for these languages are available on the Linux computers in the Computer Science lab. Results are shared in an team-delivered oral presentation. Of course, students work in teams.

Exam and Final

An exam and a final are administered. These are in-class, open-note, and open-textbook (but no other books) tests. Of course, students work on these individually.

Makeup examinations are not normally administered.

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 course score. 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, see:

[http://boisestate.atlassian.net/
wiki/spaces/LTS/pages/11436088/iClicker](http://boisestate.atlassian.net/wiki/spaces/LTS/pages/11436088/iClicker)

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, see:

www.cs.swarthmore.edu/~newhall/unixhelp/c_codestyle.html

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.

Much more information about academic integrity, including examples of academic dishonesty, is at:

<http://cs.boisestate.edu/~buff/files/www-integrity.pdf>

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 and Windows operating systems, via VMware.

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:

<https://www.boisestate.edu/coen-its/labs/lab-rules/>

Schedule

<i>Week</i>	<i>Date</i>	<i>Topic</i>	<i>Assigned</i>	<i>Due</i>	<i>Reading</i>
1	Jan 09 Mon	Introduction			1
	Jan 11 Wed				
2	Jan 16 Mon	MLK Day			
	Jan 18 Wed	Programming Language Syntax	TA1		2.0-2.1
3	Jan 23 Mon				
	Jan 25 Wed		LA1		
4	Jan 30 Mon				
	Feb 01 Wed				
5	Feb 06 Mon	Names, Scopes, and Bindings			3
	Feb 08 Wed		LA2	LA1	
6	Feb 13 Mon		IA1		
	Feb 15 Wed				4.0-4.1
7	Feb 20 Mon	Presidents' Day			
	Feb 22 Wed	Control Flow		TA1	6
8	Feb 27 Mon	Exam			
	Mar 01 Wed		IA2	IA1	
9	Mar 06 Mon				
	Mar 08 Wed	Data Types	TA2	LA2	7-8
10	Mar 13 Mon		LW1		
	Mar 15 Wed				
11	Mar 20 Mon	Spring Break			
	Mar 22 Wed	Spring Break			
12	Mar 27 Mon	Subroutines and Control Abstractions	LA3		9.0-9.4
	Mar 29 Wed			IA2,TA2	
13	Apr 03 Mon				
	Apr 05 Wed				
14	Apr 10 Mon		LW2	LW1	
	Apr 12 Wed			LA3	
15	Apr 17 Mon	LW#2 Presentations	LA4	LW2	
	Apr 19 Wed	LW#2 Presentations			
16	Apr 24 Mon	LW#2 Presentations			
	Apr 26 Wed	LW#2 Presentations		LA4	
17	May 03 Wed	Final: 2:30-4:30			