

# BALL STATE UNIVERSITY

## CS 617: Introduction to Programming

Full Fall 2023: 8/19/23 - 12/15/23

[Dr. Jennifer Coy \(Primary Instructor\)](#) | (765) 285-8641

### Course Information

#### *Ball State University Course Catalog Description*

Introduction to programming in a contemporary, mainstream, high-level programming language such as Python. Use of numeric and textual data. Use of data structures such as arrays, lists, sets, and dictionaries/maps such as those implemented by hash tables. Students are expected to be comfortable with algebraic notation as expressed in high school mathematics.

#### *Prerequisites*

This course has no prerequisites, other than you will be reminded of your high school math in some of the examples. In other words, there is no expectation of any previous programming experience – beginners are very welcome! This course is appropriate for a wide variety of students and is especially recommended for students who have not programmed before, or who are interested in learning python as a new programming language through a structured learning environment.

#### *Learning Outcomes*

This course covers a broad introduction to using programs to solve problems, introduces the student to common terms and concepts used when programming, and focuses on the python programming language. By the end of the semester, students will be able to use the python programming language to write a wide variety of programs that manipulate data and generate the desired results. Specifically, students will:

- Identify, compare, evaluate, and write programs that use the sequencing, iteration, and selection constructs in the python programming language.
- Calculate the value of expressions and explain the effect of statements in the python programming language.
- Identify, compare, evaluate, and use the formal elements of procedural abstraction including parameters, local variables, and return values.
- Write programs that use 1- and 2-dimensional data structures to represent aggregate data and iteration to compute over the structures.

#### *Course Modality and Structure*

This course is offered in an online, asynchronous format through Coursera. Course content and assignments are arranged in weekly modules. Each module is composed from the following items: recorded lectures and “live” coding examples, reading assignments, programming assignments, reflective practice assignments, short discussions, and scheduled exams. Students can work through each week’s material at their own pace.

#### *Course Time Commitment*

At Ball State University, it is expected that students will spend approximately 2 hours of study time for every one hour in class. Since this is a 3 credit hour class, you should expect to spend up to 9 hours on this class each week: approximately 3 hours of “in class” work (watching lectures and “live” coding examples, contributing to discussions, completing reflective practice work, and exams when scheduled) plus up to 6 hours per week of study time (reading assignments, writing programming assignments, and related work).

## **Course Materials**

### *Optional Textbook*

P. Deitel and H. Deitel, "Intro to Python for Computer Science and Data Science", Pearson Education, 2020. ISBN: 0-13-540467-3.

### *Computer Requirements*

A computer is required for this class, either Windows or Mac. Any modern computer should work, as all programming can be done using a browser on the Coursera platform. Program execution will occur on the Coursera servers, so all your computer needs to do is to connect through the browser. Alternately, you can choose to install the software packages directly on your computer following instructions found in the textbook (in this case, there may be minimum processor speed and memory requirements), but all assignments still need to be submitted through the Coursera interface.

## **Course Assignments**

### *Grading Summary*

<b>Assignment Type</b>	<b>Percent of Final Grade</b>
Reflective Practice	10%
Weekly Programming Assignments	25%
Programming Projects	30%
Exam 1 & Exam 2	20%
Final Exam	15%

### *Grading Scale*

A standard grading scale will be used:

100-93% A    92-90% A-    89-87% B+    86-83% B    82-80% B-  
79-77% C+    76-73% C    72-70% C-    69-67% D+    66-60% D    59-0% F

### *Description of Assignments*

#### *Ungraded Work*

Students are expected to watch the recorded lectures and "live" coding examples, as well as complete the assigned readings and other ungraded work. However, since this work is assessed through the application of this learning to other assignments, completion of this work is not directly graded.

#### *Reflective Practice*

Since it is easiest to learn how to program by *doing* programming, students are encouraged to experiment with the content covered for each week in a free-form creative activity. These activities are called *Reflective Practice* exercises, which are described in more detail on each assignment.

#### *Weekly Programming Assignments*

The content learned for each module will be applied through short programs, due each week. These assignments are designed to build on each other, and so it is important to complete them in order. Usually, two short programs will be assigned each week.

#### *Programming Projects*

Seeing it all come together into a larger project is important, and so 3 or 4 larger programming projects will be assigned. These will often have different due dates than the weekly programming assignments, allowing for more time to be spent on development.

### *Exams and Final Exam*

Two exams will be given during the semester, plus a comprehensive final exam during at the conclusion of the course. These exams will be completed online using Coursera. The exams will be released with the weekly module, and learners will have a time limit to answer the questions once the exam is started.

<b>Exam</b>	<b>Due Date</b>
Exam 1	Week 5
Exam 2	Week 10
Final Exam	Week 15

## **Course Policies and Advice**

### *Participation Policy*

- This course is designed with weekly activities, discussion, and other activities designed to build a scaffold and improve your understanding over time. The best way to learn the course material is to write frequent small programs, so be sure to keep up with the assignment schedule as much as possible.
- Be sure to log into Coursera frequently. Each module is expected to be completed in one week. Continuous participation throughout the week is much better for learning than trying to “cram” the whole week’s assignments in one sitting. Please try to spread out your work over a few days, to give your brain time to digest what it has learned.

## **University Policies**

### *Diversity Statement*

Ball State University aspires to be a university that attracts and retains a diverse faculty, staff and student body. We are committed to ensuring that all members of the campus community are welcome through our practice of valuing the various experiences and world views of those we serve. We promote a culture of respect and civil discourse as evident in our [Beneficence Pledge](#). For Bias Incident Response information, visit the [Bias Incident Reporting website](#) or email [mc2@bsu.edu](mailto:mc2@bsu.edu).

### *Disability Services*

If you need course adaptations or accommodations because of a disability, please contact Disability Services as soon as possible. The [Office of Disability Services](#) coordinates services for students with disabilities; documentation of a disability needs to be on file in that office before any accommodations can be provided. Disability Services can be contacted at 765-285-5293 or [dsd@bsu.edu](mailto:dsd@bsu.edu).

### *Attendance Policies*

Faculty are required to establish attendance policies for their courses and ensure that they communicate these policies through their course syllabi. In some cases, faculty may be asked to provide the last date of attendance for a student in association with financial aid requirement. Students are expected to review course syllabi regarding absence guidelines and follow those guidelines. Course attendance policies must be consistent with University policy. The University has a number of specific policies regarding student absences that are housed within different areas. [Explore Ball State course attendance policies](#). For CS 617, attendance is defined as regularly participating in the weekly module activities.

### *Ball State Academic Ethics Policy*

Honesty, trust, and personal responsibility are fundamental attributes of the university community. Academic dishonesty and other forms of academic misconduct threaten the foundation of an institution dedicated to the pursuit of knowledge and will not be tolerated. To maintain its credibility and reputation, and to equitably assign evaluations of scholastic and creative performance, Ball State University is committed to maintaining a climate that upholds and values the highest standards of academic integrity. Visit the complete [Student Academic Ethics Policy](#).

### *Plagiarism Policy*

Plagiarism is not acceptable. You are probably familiar with how plagiarism applies to written work, such as essays. For a programming class, plagiarism is similar – your “essay” is your code. There are many different ways to successfully solve each programming problem, and so code that is independently written is usually different than other work submitted by your classmates. You can discuss with each other in general terms how to approach solving a problem, but should never share specific code, either verbally or in written form. Unless otherwise indicated for all assignments, you must work independently by yourself. Never email or directly share your code with other students, post it online, or otherwise disseminate solutions to assignments.

There are often questions about how plagiarism translates to programming, so we will discuss this in class as well. Please ask questions if you are uncertain about what is acceptable and what is not.

#### *Potential Violations*

Examples of academic integrity violations include but are not limited to:

- Working with another person on any assignment other than authorized group projects.
- Sharing or allowing others to access your files, whether done with permission or not.
- Use or possession of a file created by someone else.
- Reusing work from another semester, course, or section.
- Soliciting others to complete work for you.

### **Academic Support Services**

*Need to insert information regarding Academic Support for students.*

### **Subject to Change Statement**

This syllabus and schedule are subject to change in the event of extenuating circumstances.