



CS 263: Syllabus (Fall 2026 - Second 8 Weeks)

Thomas J. Kennedy

Last modified: Jan 6, 2026

Contents:

[1 Basic Course Information](#)

[1.1 Catalog Description](#)

[1.2 Overall Description](#)

[1.3 Prerequisites](#)

[1.4 Instructor](#)

[2 Required Materials](#)

[2.1 Textbooks](#)

[2.2 Supplemental Materials](#)

[2.3 Technology Requirements](#)

[3 Course Objectives](#)

[4 Course Schedule](#)

[5 Grading](#)

[5.1 Assignments](#)

[5.2 Exams](#)

[5.3 Incomplete Grades](#)

[6 Expectations](#)

[6.1 Assignment Expectations](#)

[7 Course Policies](#)

[7.1 Due Dates and Late Submissions](#)

[7.2 Academic Honesty](#)

[8 University Policies](#)

[8.1 Code of Student Conduct and Academic Integrity](#)

[8.2 Honor Pledge](#)

[8.3 Discrimination Policy](#)

[8.4 Diversity and Inclusion](#)

[8.5 University Email Policy](#)

[8.6 Withdrawal](#)

[8.7 Privacy of Student Information](#)

[8.8 Educational Accessibility](#)

1 Basic Course Information

The course schedule and website are located at <https://www.cs.odu.edu/~tkennedy/cs263/latest>.

1.1 Catalog Description

Laboratory work required. An introduction to the Python programming language for students who are familiar with programming in C++ or Java. Topics include basic language syntax, data structures, control flow, classes, inheritance, and basic elements of the Python standard library. Not open to students with credit for CS 253.

1.2 Overall Description

This course is intended as an introduction to Python for students already familiar another language (with an emphasis on C++ or Java). The overall focus of this course is to prepare to teach you how to apply your existing knowledge to write *Pythonic* code.

1.3 Prerequisites

The prerequisites for this course are:

- CS 250, Problem Solving and Programming II in C++, or CS 251, Programming and Problem Solving in Java
- [CS 252](#), Introduction to Unix for Programmers

1.3.1 General Programming Knowledge

Students should be familiar with certain basic programming techniques that are largely independent of any specific programming language:

- using editors, compilers and other basic software development tools.
- basic software design (i.e., stepwise refinement and top-down design)
- software testing, including the use of scaffolding code (stubs and drivers), selection of test cases for black-box testing, and head to head testing.

1.3.2 C++ & Java Knowledge

I will assume that you are familiar with the basics of C++ and/or Java. This includes:

- the various C++/Java statements and control-flow constructs,
- the built-in data types,
- the use of arrays, pointers, pointers to arrays, and linked lists,
- the use and writing of functions, and
- the basic use of structs and/or classes for implementing abstract data types.

1.3.3 Python 3 & Rust Knowledge

Prior knowledge of [Python](#) and [Rust](#) is neither expected nor assumed.

1.3.4 General Computer Literacy

You will be studying techniques in this course for preparing professional-quality software documentation. The key embedded word in “software documentation” is “document”. Students taking this course should be able to use word processors and other common tools to produce good quality documents, including mixing text and graphics in a natural and professional manner.

1.4 Instructor

General Information	
Instructor	Thomas J. Kennedy
Office	Monarch Hall 2123G
Phone #	757.683.7725
Email	tkennedy@cs.odu.edu
Home Page & Office Hours	https://www.cs.odu.edu/~tkennedy

Important: All email related to this course should have the phrase **CS 263** somewhere in the subject line. This flags your message in my mailbox for faster attention.

I try to respond to all (properly marked) messages

- before I leave campus each day (Monday through Friday)
- within 48 on weekends & holidays.

This is my general email policy. However, this is usually $O(n)$ (i.e., an upper bound).

Any delays in replying to email will be noted in a Canvas Announcement.

1.4.1 Office Hours

My general office hours are available at <https://www.cs.odu.edu/~tkennedy/>. Instructions for scheduling a formal appointment are listed on the same page and [here](#). My office hours consist of web-conference (Zoom) appointments.

General questions about course content and reports of website problems should normally be asked in the Forums on Canvas or via email.

Questions about grades, how to solve assignments and other graded activities must be send to tkennedy@odu.edu.

For more discussion on course communications, please refer to the [Communications policy](#).

2 Required Materials

2.1 Textbooks

There is no required textbook.

2.2 Supplemental Materials

All supplemental resources and materials will be available in Canvas.

2.3 Technology Requirements

2.3.1 Computer Accounts

Students will need an account on the CS Dept. Linux network to participate in this class. This account is unrelated to any University-wide account you may have from ODU's Information Technology Services (ITS).

If you have had a CS account in the recent past, you should find it still active with your login name, password, and files unchanged. If you have had an account and it has not been restored, contact the CS Dept systems staff at root@cs.odu.edu requesting that it be restored.

If you do not yet have such an account, go to the [CS Dept. home page](#) and look for “Account Creation” under “Online Services”. All students in this course are responsible for making sure they have a working CS Unix account **prior** to the first assignment.

2.3.2 Compilers and Interpreters

The “official” environment in which students’ programming assignments will be evaluated is defined by our Dept. Linux servers. It is the student’s responsibility to be sure that their code compiles and executes using the compilers and run-time environment provided there. As of this writing, the compiler and run-time environment versions used are...

- C++: g++ 11.4.0
- Java: openjdk version “21.0.8”
- Python: 3.11 & 3.13

You may want to install [uv](#). It will handle the creation of Python virtual environments and the installation of any required libraries. Most Python examples can be run with...

```
uv run [program].py
```

...in place of...

```
python3.11 [program].py
```

3 Course Objectives

The overall goal of this course is to *prepare students to write, design, refactor, and test code in Python at a level comparable to a student completing a 200-level programming course in C++ (CS250), Java (CS 251), or Python (CS 253) in preparation for 300-level and 400-level coursework.*

A student who successfully completes this course will be able to (in Python):

1. Run a program consisting of a single file and containing a main function.
2. Run a program consisting of multiple modules and containing a main function.
3. Organize code into multiple modules.
4. Write tests for a module.
5. Apply the basics of test-driven development through PyTest and/or unittest.
6. Make use of the various loops (for and while)
 - A. Compare the various loops (for and while)
 - B. Choose the most appropriate loop (for or while) for a given problem
7. Make use of the conditional blocks (i.e., if, if-else, and if-else-if-else).
 - A. Compare the various conditional blocks (i.e., if, if-else, and if-else-if-else)
 - B. Construct the appropriate conditional block (i.e., if, if-else, and if-else-if-else) for a given problem.
8. Test and write functions.
9. Design ADTs in accordance with the Class Checklist.
10. Discuss when polymorphism is appropriate.
11. Discuss when it is appropriate to utilize dataclasses, classes, and enums.
12. Write code that utilizes dunder functions.
13. Refactor code to follow best practices (e.g., PEP 8 and PEP 20).
14. Apply code linting tools (e.g., pylint and black) to write idiomatic (Pythonic) code.
15. Discuss the various NumPy `np.array` mechanics (e.g., broadcasting).

4 Course Schedule

The course schedule is available in Canvas under *Modules* and *Calendar*. An alternate schedule/outline can be accessed at <https://www.cs.odu.edu/~tkennedy/cs263/latest/Directory/outline/index.html>.

5 Grading

Your Final Course Grade will be computed using the following weights:

Assignments	60%
Quizzes	30%
Final Exam	10%

The Final Exam is cumulative.

Example 1: Grading Schema

Percent Letter Grade 4pt Value

≥ 85	A	4.0
≥ 80.5	A-	3.7
≥ 74.5	B+	3.3
≥ 70	B	3.0
≥ 60	B-	2.7
≥ 59.5	C+	2.3
≥ 55	C	2.0
≥ 50.5	C-	1.7
≥ 44.5	D+	1.3
≥ 40	D	1.0

Percent	Letter Grade	4pt Value
N/A	D-	0.7
< 39	F	0.0

5.1 Assignments

Assignments for this course will include *programming assignments* (in Python) which must be done on an individual basis.

5.1.1 Assignment Grading

Assignments will be turned in through a combination of the CS submission system or through Canvas. Each assignment will include submission instructions.

Unless the assignment explicitly states otherwise, you may submit a total of three times per assignment; the instructor will take the last of the marks, although you may request that your score be “rolled back” to an earlier one. You may **not** submit after viewing the sample solution (if one is provided).

Most of the assignments will be graded by an automatic grader. The results will be sent to your ODU email account.

5.1.2 Auto-Grader & Testing

Test driven development is a topic of particular import—not only in academia, but in industry.

You will be expected to make use of Blackbox Testing. This is a topic of particular import. Blackbox testing is covered in CS 251—a prerequisite for this course. You will also need to make use of white-box testing and unit testing.

All tests are designed by me—the instructor. The Auto-Grader runs tests that I use to evaluate my solution. These tests evaluate mechanics of import—e.g., dynamic binding and function overloading.

Be systematic in all changes to your assignment solution and modifications to your tests. Do not haphazardly make changes to an assignment and resubmit hoping for a better grade. Treat each submission attempt as your final submission. Ask for guidance before each subsequent submission.

5.2 Exams

Exams will be administered online via Canvas.

5.3 Incomplete Grades

A grade of “I” indicates assigned work yet to be completed in a given course, or absence from the final examination, and is assigned only upon instructor approval of a student request. The “I” grade may be awarded only in exceptional circumstances beyond the student’s control. The “I” grade becomes an “F” if not removed by the day grades are due for following term based on specific criteria: [Incomplete, Withdraws and Z grades.](#)

6 Expectations

6.1 Assignment Expectations

It is my expectation that you have completed approximately 70% of each assignment once half the allotted time has passed. For a two week assignment this would be one week. This will allow you sufficient time to address any issues, refine your testing process, and discuss your solution with me during my office hours.

I expect every student to discuss each assignment with me at least once.

7 Course Policies

7.1 Due Dates and Late Submissions

Late papers, assignments, projects, and make-up exams will not normally be permitted.

Exceptions will be made only in situations of unusual and unforeseeable circumstances beyond the student's control, and such arrangements must be made prior to the due date in any situations where the conflict is foreseeable.

"I've fallen behind and can't catch up", "I'm having a busier semester than I expected", or "I registered for too many classes this semester" are *not* grounds for an extension.

Extensions to due dates will not be granted simply to allow "porting" from one system to another. *"But I had it working on my home PC!"* is not an acceptable excuse.

7.2 Academic Honesty

Everything turned in for grading in this course must be your own work. If an assignment is **explicitly** described as a team assignment, it must be the work of the team members only.

The instructor reserves the right to question a student orally or in writing and to use his evaluation of the student's understanding of the assignment and of the submitted solution as evidence of cheating. Violations will be reported to the Office of Student Conduct & Academic Integrity for consideration for possible punitive action.

Students who contribute to violations by sharing their code/designs with others may be subject to the same penalties.

- Students are expected to use standard Unix protection mechanisms (chmod) to keep their assignments from being read by their classmates. Failure to do so will result in grade penalties, at the very least.

This policy is *not* intended to prevent students from providing legitimate assistance to one another. Students are encouraged to seek/provide one another aid in learning to use the operating system, in issues pertaining to the programming language, or to general issues relating to the course subject matter.

Students should avoid, however, explicit discussion of approaches to solving a particular programming assignment, and under no circumstances should students show one another their code for an ongoing assignment, nor discuss such code in detail.

Use of Online Resources

You may **not** post details of course assignments, projects, or tests at online Forums, Bulletin Boards, Homework sites, etc., soliciting help.

You may use information that you have not solicited but have located, subject to the following restrictions:

- Just as when writing a paper, if you use someone else's ideas, you must cite your sources appropriately. Within code, such citations appear in comments.

Example:

```

:
double x = 23.0;
double xsqrt = sqrt(x);
// Search algorithm based upon code by S Zeil at
// https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#orderedsequ
int loc = 0;
while (loc < arraySize && numbers[loc] < xsqrt)
:

```

- Just as when writing a paper, if you use someone else's words (code), you must cite your sources appropriately **and** mark the quoted text. Within code, such citations appear in comments.

Example:

```

:
double x = 23.0;
double xsqrt = sqrt(x);

// Begin quoted code from S Zeil at
// https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#orderedsequ
int loc = 0;
while (loc < listLength && list[loc] < searchItem)
{
    ++loc;
}
// End quoted code

:

```

- Failure to appropriately cite any such “found code” will be taken as evidence of plagiarism.
- The overall principle stated in the first sentence of this section remains in effect. “Everything turned in for grading in this course must be your own work.” If the bulk of your assignment, project, test answer, etc., are copied, even with appropriate citation, to the degree that, in the judgment of the instructor, you have not demonstrated your own knowledge of the course material, you will receive a zero for that submission.

8 University Policies

8.1 Code of Student Conduct and Academic Integrity

The Office of Student Conduct & Academic Integrity (OSCAI) oversees the administration of the student conduct system, as outlined in the Code of Student Conduct. Old Dominion University is committed to fostering an environment that is: safe and secure, inclusive, and conducive to academic integrity, student engagement, and student success. The University expects students and student organizations/groups to uphold and abide by standards included in the [Code of Student Conduct](#). These standards are embodied within a set of core values that include personal and academic integrity, fairness, respect, community, and responsibility.

8.2 Honor Pledge

By attending Old Dominion University, you have accepted the responsibility to abide by the Honor Pledge:

I pledge to support the Honor System of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community it is my responsibility to turn in all suspected violations of the Honor Code. I will report to a hearing if summoned.

8.3 Discrimination Policy

Rewritten to actually make sense. The original indicated that *discrimination* was defined below, but did not include a definition.

The purpose of this policy is to establish uniform guidelines in order to promote a work and education environment that is free from harassment and discrimination, as defined in [ODU's Discrimination Policy](#), and to affirm the University's commitment to foster an environment that emphasizes the dignity and worth of every member of the Old Dominion University community.

[ODU's Discrimination Policy](#) details the process to address complaints or reports of retaliation.

8.4 Diversity and Inclusion

The [Division of Student Engagement & Enrollment Services](#) values the uniqueness of our Monarch community. The word “engagement” reflects our commitment to embrace the differences in our cultural backgrounds, perceptions, beliefs, traditions, world views, socio-economic status, cognitive and physical abilities.

We will strive to serve as the pre-eminent model for engaging every student to achieve their own success. Our core values are fueled by our responsibility and actions toward community development and engagement, cultural competence and understanding, physical and mental wellness and inclusion for every member of ODU. We will embrace our greatest strength - the diverse composition of our student body and workforce.

For more information regarding diversity and inclusion, please visit the Office of [Intercultural Relations](#).

8.5 University Email Policy

Reformatted to follow <https://www1.odu.edu/about/policiesandprocedures/computing/standards/11/02>.

8.5.1 Student Email

With the increasing reliance and acceptance of electronic communication, email is considered an official means for University communication. Old Dominion University provides each student an email account for the purposes of teaching and learning, research, administration, and service. It is important that all students are aware of the expectations associated with email use as outlined in the [Student Email Standard](#).

8.5.2 Email Account Activation

It is the responsibility of every eligible student to activate MIDAS, the Monarch Identification and Authorization System, in order to obtain email access.

8.5.3 Expectations Regarding Use of Email

The email account provided by the University is considered to be an official point of contact for correspondence. Students are expected to check their official e-mail account on a frequent and consistent basis in order to stay current with University communications. Mail sent to the ODU email address may include notification of University-related actions, including academic, financial, and disciplinary actions. For more information about student email, please visit [Student Computing](#).

8.5.4 Educational Uses of Email

University offices and instructors cannot validate that a communication coming by email is from an ODU student unless it comes from a valid ODU email address. If students send mail from non-ODU email accounts (e.g., Hotmail or Yahoo), faculty and staff are not obligated to respond and may request that official e-mail accounts be used.

8.6 Withdrawal

Enrollment in this course indicates your acceptance of its teaching focus, requirements, and policies. Please review the syllabus and the course requirements as soon as possible. If you believe that the nature of this course does not meet your interests, needs or expectations, if you are not prepared for the amount of work involved – or if you anticipate assignment deadlines or adherence to course policies will constitute an unacceptable hardship for you – you should drop the course by the drop/add deadline, which is listed in the [ODU Schedule of Classes](#). Visit the [Office of the University Registrar](#) for more information.

8.7 Privacy of Student Information

Old Dominion University recognizes its duty to uphold the public's trust and confidence, not only in following laws and regulations, but in following high standards of ethical behavior. Members of the Old Dominion University community are responsible for maintaining the highest ethical standards and principles of integrity. The [Code of Ethics](#) is a set of values-based statements that demonstrate the University's commitment to this goal. The [Privacy of Student Information](#) details Family Educational Rights & Privacy

Act (FERPA), along with other information regarding privacy.

8.8 Educational Accessibility

Old Dominion University is committed to ensuring equal access to all qualified students with disabilities in accordance with the Americans with Disabilities Act. The Office of Educational Accessibility (OEA) is the campus office that works with students who have disabilities to provide and/or arrange reasonable accommodations.

- If you experience a disability which will impact your ability to access any aspect of my class, please present me with an accommodation letter from OEA so that we can work together to ensure that appropriate accommodations are available to you.
- If you feel that you will experience barriers to your ability to learn and/or testing in my class but do not have an accommodation letter, please consider scheduling an appointment with OEA to determine if academic accommodations are necessary.

The Office of Educational Accessibility is located at 1021 Student Success Center and their phone number is (757) 683-4655. Additional information is available at the [OEA website](#)



© 2015-2026, Old Dominion Univ.