EC327 Introduction to Software Engineering

Syllabus

Summer 2025

Administration

Times/Places

The lecture is Monday thru Thursday 1-3pm in PHO 117.

This is a summer one course which starts on Tuesday May 20 and completes on Thursday June 26th.

Contact Information

Outside of class, the best way to contact Dr. Carruthers is via the course google chat.

You can also text him at 617-264-7939.

Please always include "ec327" in the subject line of an email or at the start of the text.

Course content

This course aims to introduce software design, programming techniques, data structures, and software engineering principles. Computer hardware and organization. Machine and assembly languages. Fundamental elements of functional programming languages. Principles of object-oriented programming. Introduction to elementary data structures and algorithmic analysis. Core competencies in software engineering, including programming style, optimization, debugging, compilation, program management, development environments, software repositories, version control.

References

- The Rust Programming Language "The Book"
- TutorialsPoint Rust

- Rustlings
- The Rust Reference
- Rust By Example

Online Resources

The course materials and submission system are available here: EC327 Course Website

Grading and Assessment

Collaboration policy

The University has very clear policies regarding academic honesty. It considers plagiarism and other forms of cheating serious offenses and will enforce serious penalties when they occur. All students are required to abide by all applicable policies and regulations on academic honesty.

Here is the academic conduct code for Boston University

Grading Components

Component	Weight
Weekly Projects	100%

Expectations of Achievement Levels

A-level student

This student consistently delivers A-level software, which means the software

- is well designed
- is well documented
- is conformant to style and formating guidelines
- achieves all customer requirements
- is delivered on time

The student takes an active and positive role in the class community, often also involving helping other students, discovering specifications that need clarification

Please note that achieving an A is very difficult and is reserved for students with truly exceptional performance in all aspects of the course. Typically, three times as many A- grades are awarded as full A grades.

B-level student

This student sometimes achieves the standards for A-level software, and when that level is not achieved, most other software delivered is B-level, which means the software

- does not contain serious design flaws
- is mostly conformant to style and formating guidelines
- achieves primary customer requirements (i.e. basic functionality, but fails checks)

The student

- takes an active and positive role in the class community, and
- demonstrates effort and commitment to improving their software design techniques

C-level student

This student rarely achieves A-level software (usually only on the simplest of tasks), and sometimes achieves B-level software.

The rest of the software delivered is C-level software, which means the software is in one of three stages

- compiles and meets some customer requirements
- compiles but meets no other customer requirements, instead having pseudocode/shell programs
- does not compile, has logic or design flaws, but is otherwise "complete"

This student

- demonstrates effort and commitment to improving
- demonstrates basic understanding of fundamentals of software development: syntax, algorithms, program organizaiton
- is able to design algorithms/pseudocode for incomplete or incorrect software
- takes a reasonably active role in the class community (i.e. comes to class most of the time, submits questions on discussion board)

D or F level student

This student rarely achieves A-level or B-level software, and often submits nothing at all or just bare sketches of programs.

This student

- never demonstrates basic knowledge of syntax, program organization, or good design principles
- does not demonstrate commitment to learning
- does not participate to a satisfactory level in class activities