

e2020 Curriculum Briefing Computer Science II (EL5723)

Course Description

In this course, students will build on the computer science and programming skills gained in Computer Science I. Students will continue to learn and apply key computer science concepts to programs created in and run by Python. This course also includes questionnaires with people who use programming in their careers, providing students an opportunity to learn about the real-world experiences of programmers. The experience students will gain through completing a wide range of hands-on projects will continue to develop the fundamental skills they will need to get started as a programmer. By the end of this two-semester course sequence, students will gain many of the fundamental skills and conceptual background needed to get started as a programmer in Python or in any programming language. Project-based learning walks students through writing code, while course assignments build on what students learn.

This course is aligned with the Washington State CTE Computer Programming framework and the Texas State TEKS Computer Programming framework.

Topics of Study

- List manipulation
- Creating interactive drawing programs
- Loops and nesting
- String formatting
- Programming menus
- Creating text adventure games
- Creating dice games

e2020 Curriculum Briefing (continued)

Course Science II (EL5723)

Course Features

- While explaining concepts and code syntax to support student learning, the course provides step-by-step instruction on how to write code.
- The course uses Python's built-in, integrated development environment, IDLE, for writing code. IDLE provides a command line programming environment for testing individual lines of code, as well as a text editor for writing longer programs.
- The Python programming language's lack of a compiling requirement speeds up the time it takes for students to run and test code.
- The course uses Test Your Code interactive exercises that allow students to practice fixing coding errors.
- Illuminating course graphics provide another way to understand complicated course material.
- Course assignments guide students in improving code they've already written. Other assignments present students with error-ridden code and have them fix the errors in the code. A third type of assignment presents students with a programming challenge and gives instructions to solve that challenge by writing a new program.
- Personal questionnaires with programming professionals provide insight into the ways in which programming skills can be used in different careers.

Grading

Just as with our other e2020 courses, you always have the flexibility to tailor the grade weights for the course according to your own district, school, and student needs.

The default grades for this course are as follows:

- | | |
|---------------|-----|
| • Assignments | 20% |
| • Essays | 0% |
| • Quizzes | 20% |
| • Tests | 10% |
| • Exams | 0% |
| • Projects | 50% |
| • Additional | 0% |