

STRANDS AND STANDARDS

COMPUTER PROGRAMMING CAPSTONE



Course Description

This course is the culminating experience for students in computer programming. It provides an opportunity for students to integrate and apply their knowledge and skills acquired throughout the program to a real-world problem or an innovative product idea. Students will work in teams to design, develop, test, and evaluate a software solution that meets the needs and expectations of a client from industry or society. The course will also foster students' professional skills such as communication, collaboration, project management, and ethical reasoning. Students will present their work regularly to the instructor, the client, and their peers, and receive feedback and guidance. The course will require students to demonstrate creativity, critical thinking, and problem-solving abilities, as well as technical competence and proficiency in their chosen programming language and tools.

Intended Grade Level	11-12
Units of Credit	1.0
Core Code	35.02.00.00.044
Concurrent Enrollment Core Code	None
Prerequisite	Computer Programming 2 or Advanced
Skill Certification Test Number	Capstone Rubric
Skill Certification Cut Score	N/A
Test Weight	1.0
License Area of Concentration	CTE and/or Secondary Education 6-12
Required Endorsement(s)	
Endorsement 1	Programming & Software Development

STRAND 1

Students will understand and complete a formal software development lifecycle to address a specific design problem.

Standard 1

Understand the ordered software development life cycle.

- Requirements Analysis: Identify specifications and understand requirements to create a solution to a problem
- Planning/Design: Design an algorithm to solve the problem using appropriate documentation (UML diagrams and pseudocode). UML Design = UML (Unified Modeling Language) Design.
 - Define an algorithm
 - Break the problem down into its subcomponents using top-down design
- Implementation: Write the code, with comments, to implement the algorithm
- Testing: Test program for verification of errors and proper functionality
- Release and Maintenance: Release the solution and provide updates when necessary

Standard 2

Identify the Problem/Solution

- Identity and define the problem
- Define several possible solutions
- Narrow and decide on the most optimal solution
- List requirements
- Identify constraints

Standard 3

Design a solution with a Unified Modeling Language (UML) activity diagram and class diagram as needed

- Create an activity diagram
- Create a class diagram for the class hierarchy of a program
- Create a sequence diagram for a method
- Translate diagrams to code

Standard 4

Utilize a project management system process:

- Agile
- Scrum
- Waterfall

Standard 5

Present and refine project design based on client feedback

- Create a meeting agenda
- Complete a formal pitch to the client
- Meet with client for feedback and discussion
- Modify design as needed.

Standard 6

Continuous improvement on project based on alpha and beta test users

- Communicate professionally with experts and mentors to obtain feedback on the technical feasibility of the design, document the interactions, and implement recommended changes.

Standard 7

Delivering the project to the client

- Students will put together a formalized package of the product, including thorough user-friendly documentation for the client to use and review.
- Complete a comprehensive, multimedia presentation and portfolio that provides an overview of each step of the design experience using a variety of media.
- Students will present and showcase their finalized product, include all implemented features, to a committee including a teacher and the client.

STRAND 2

Project Requirements:

- Based on the teacher's recommendation, students will be placed on a team.
- The team will include a teacher advisor.
- The team will have a client/liaison with a sponsoring company.
 - In the case that no sponsoring company can be found, the sponsoring company/client can be the school or the school district.
- The team will meet with the client to discuss their software needs.
- The team will draft an official project proposal
- The team will collaborate and present a formal project proposal to a committee consisting of their client and teacher.
 - In this presentation, students will describe a formal software development solution for a specific problem/need of the client.
- The project will scale for a full year (the duration of the course).
- Students will use Agile/SCRUM project management system to manage their projects.
- Students will utilize source control to share the project amongst team members.
 - Students will explore different source/version control options for software development, weigh the pros and cons and select the best system for their team and project.
 - Students will analyze different language/platforms and their considerations in selecting the best one for the project.
- Students will provide a formal progress report to their teacher as determined by the project lead
- Students will present their progress to the client in regular meetings determined by the project lead
 - Students will utilize professional skills to communicate with clients and create meeting agendas
- Students will participate in a performance review with their teacher each term in which progress/grades will be determined.
- The final product will be deliverable and usable by the company, including properly documented user-friendly documentation for the client to use and review.

STRAND 3

Students will participate in one or more of the following student competitions or leadership activity.

Standard 1

Participate in a computer programming related student competition (these are a few of the most common options, please see your teacher for additional opportunities)

- Enter a school, district, or state level computer programming related competition
- Enter and compete in a CTSO competition in a computer programming related competition

Standard 2

Participate in a computer programming related student leadership opportunity.

- Participate in a CTSO Leadership opportunity
- Arrange a job shadow opportunity with a computer programming related professional and observe their daily tasks, skills, and challenges. Ask relevant questions and record notes on the experience. Write a reflection paper on what you learned and how it relates to your career goals.
- Identify and contact a computer programming related professional who is willing to share their expertise and insights with you. Prepare a list of interview questions that cover topics such as their education, training, work experience, current projects, challenges, and advice. Conduct the interview either in person, by phone, or online, and record their responses. Write a report that summarizes the main points of the interview and reflects on how this experience has influenced your career aspirations and plans.

Overall Performance Skill

Students will use the Strands & Standards in this course to create a fully functional program that solves a specific problem.

Workplace Skills

Workplace Skills taught:

- Communication
- Problem Solving
- Teamwork
- Critical Thinking
- Dependability
- Accountability

Performance Skills Documentation & Rubric:

A link to the performance skill rubric is located [HERE](#). It is also located adjacent to the Strands & Standards document online.

For each student:

- Print a copy of the rubric.
- Complete the rubric based on the student's performance.
- Compile all completed rubrics into a single PDF document.

Once compiled, upload the final PDF to **YouScience** for submission.