NEW JERSEY CENTER
FOR TEACHING & LEARNING

Progressive Science Initiative® (PSI®)
CSCI6347: Learning and Teaching AP Computer Science A Part I

Primary Student Contact: Maria Surace   maria@njctl.org

Faculty Team: Dr. Bob Goodman   bob@njctl.org
                               Maria Surace   maria@njctl.org
                               Katy Goodman   katy@njctl.org

Course Credit: 4.0 NJCTL credits

Dates & Times:
This is a 4-credit, self-paced course, covering 5 modules of content. The exact number of hours that you can expect to spend on each module will vary based upon the module coursework, as well as your study style and preferences. You should plan to spend approximately 15 hours per credit working online, and up to 30 hours per credit working offline.

Graduate Student Handbook: www.njctl.org/graduate-handbook/

COURSE DESCRIPTION:
This course is for teachers to learn the content of *PSI Advanced Placement Computer Science A* and how to teach that course to students. It focuses on fundamental programming skills and thought processes required for successful programming in any language while integrating components of Java.

Topics include Fundamentals of Programming; Control Statements & Loops; Methods & Arrays; and Strings.

STUDENT LEARNING OUTCOMES/OBJECTIVES:
Upon completion of the course, the student will be able to:

1. Demonstrate proficiency in programming in Java, as listed in the module learning outcomes below.
2. Integrate PSI materials (including presentations, labs, practice problems, etc.) to support student learning and deliver effective instruction.
3. Create a social constructivist learning environment through the use of formative assessment questions, interpreting the results of this assessment to effectively facilitate student-led discussions that support deeper understanding of the content.
4. Integrate multiple attempts to demonstrate student mastery of content knowledge, as encouraged/fostered by the PSI pedagogy.

5. Implement learning plans that are aligned to AP standards, incorporate literacy strategies and allow for differentiation.

TEXTS, READINGS, INSTRUCTIONAL RESOURCES:
Required Texts:
- PSI AP Computer Science A uses a free digital textbook accessible at: https://njctl.org/materials/courses/ap-computer-science-a/

- Participants will download SMART Notebook presentations, homework files, labs, and teacher resources from the PSI AP Computer Science A

COURSE REQUIREMENTS:
In order to receive a Passing grade, the participant must complete the following course requirements:

1. Activities: A number of different learning activities will ensure participant engagement and learning in the course. These include:
   - Engage in video module lessons which demonstrate minimized direct instruction followed by frequent formative assessment
   - Completion of formative assessments aligned to learning objectives which include detailed analysis when answered incorrectly.
   - Interaction with module discussion boards that allow conversation with peers and course instructors about the module's content, delivering that content to students. Discussion boards also serve as a place to ask and answer questions related to the module's content.
2. Short Answer Assignment: Each module requires one (1) original response to a given prompt. These prompts are typically based upon course lessons and require teachers to analyze, reflect, and make connections between the module's content and their own classroom practice.
3. Mastery Exercises: For each module, these multiple-choice question quizzes assess the content knowledge gained in a module. Participants have the opportunity to retake; random questions are pulled from a larger question bank on each attempt ensuring varied questions.
4. Programming Assignments/Labs: In each module, a .java file is submitted to demonstrate an understanding of the graded programming assignment. These assignments are problem-based questions that require the writing of a program. These promote a deeper understanding of logical reasoning and the applications of programming. A culminating lab activity is embedded into the final module.
5. Module Exam: One is completed at the end of each module. It is a culminating exam consisting of multiple choice and free response questions aligned to the standards and objectives of the module.
6. Reflection Paper: At the end of the course, participants are required to reflect on the knowledge taught in the course, make connections, and compare/contrast their current pedagogy with new strategies gained in this assignment.
7. Final Exam: At the end of the course, a comprehensive exam consisting of Multiple Choice and Free Response questions assesses the content knowledge learned throughout the course and aligns to the AP College Board Exams.

GRADE DISTRIBUTION AND SCALE:
Grade Distribution:

Module Exams	70%
Final Exam	10%
Programming Assignments/Labs	6%
Short Answer Assignments	6%
Mastery Exercises	6%
Reflection Paper	2%

Grade Scale:

A	93 – 100
A-	90 – 92
B+	86 – 89
B	83 – 86
B-	80 – 82
C+	77 – 79
C	73 – 76
C-	70 – 72
D	60.0 – 69.9
F	59.9 or below

ACADEMIC STANDING:
NJCTL has established standards for academic good standing within a student's academic program. Students enrolled in any NJCTL online course must receive an 80 or higher to successfully complete a course and receive credit for that course. An 80 is equivalent to a GPA of 2.7 or B-. Additionally, students in an endorsement program must receive a cumulative GPA of 3.0 for all courses combined in order to successfully complete the program.

ACADEMIC INTEGRITY:
Students must assume responsibility for maintaining honesty in all work submitted for credit and in any other work designated by the instructor of the course. Academic dishonesty includes cheating, fabrication, facilitating academic dishonesty, plagiarism, reusing /repurposing your own work, unauthorized possession of academic materials, and unauthorized collaboration.

CITING SOURCES WITH APA STYLE:
All students are expected to follow proper writing and APA requirements when citing in APA (based on the APA Style Manual, 6th edition) for all assignments.

DISABILITY SERVICES STATEMENT:
We are committed to providing reasonable accommodations for all persons with disabilities. Any student with a documented disability requesting academic accommodations should contact the Dean of Students, Melissa Axelsson, for additional information to coordinate reasonable accommodations for students with documented disabilities (melissa@njctl.org).

NETIQUETTE:
Respect the diversity of opinions among the instructor and classmates and engage with them in a courteous,

respectful, and professional manner. All posts and classroom communication must be conducted in accordance with the student code of conduct. Think before you push the Send button. Did you say just what you meant? How will the person on the other end read the words?

Maintain an environment free of harassment, stalking, threats, abuse, insults or humiliation toward the instructor and classmates. This includes, but is not limited to, demeaning written or oral comments of an ethnic, religious, age, disability, sexist (or sexual orientation), or racist nature; and the unwanted sexual advances or intimidations by email, or on discussion boards and other postings within or connected to the online classroom.

If you have concerns about something that has been said, please let your instructor know.

CLASS SCHEDULE:

Module	Module Learning Outcomes	Assignments
Course Resources	● AP CS A Course & Exam Description	● Short Answer Assignment
1 – Fundamentals of Programming	● Understand computer basics, programs and operating systems. ● Create and compile simple Java programs to display an outcome on a console. ● Explain the difference between syntax, runtime and logic errors. ● Explain the basic syntax and documentation of a Java program. ● Write programs to perform simple computations using assignment statements and assignment expressions. ● Identify and name variables and constants correctly. ● Explore primitive data types and input data. ● Perform operations using basic and exponential operators. ● Perform basic string concatenation. ● Understand how content is implemented in a student-centered classroom as it applies to logic and algorithmic design.	● Short Answer Assignment ● Graded Programming Assignment ● Mastery Exercises ● Module Exam
2 – Control Statements & Loops	● Select and implement operators as conditions within control statements and loops. ● Declare, initialize and organize variables within loops. ● Create control statements and loops to provide the ability for varied outcomes within a program by selecting which code segments run ● Explain the differences between syntax errors, runtime errors, and logic errors. ● Analyze and evaluate variables within a loop compared to those outside the loop. ● Write programs to perform simple computations using control statements and loops. ● Generate random numbers and use various other Math Library commands. ● Understand the similarities and differences between types of loops and how to vary their use.	● Short Answer Assignment ● Graded Programming Assignment ● Mastery Exercises ● Module Exam

3 – Strings	• Select and implement appropriate built in string methods to perform key tasks when writing programs. • Create strings to increase the versatility and complexity when writing programs. • Create strings to increase the versatility and complexity when writing programs. • Understand the existence of variables and how they are affected through various actions throughout the structure of a program. • Use various commands from the math library in the context of a program • Examine and debug compiler and runtime errors within programs. • Format output to a console.	• Short Answer Assignment • Graded Programming Assignment • Mastery Exercises • Module Exam
4 - Arrays	• Select and implement appropriate built in array methods to perform key tasks when writing programs. • Create arrays to increase the versatility and complexity when writing programs. • Understand the elements of an array and how to create, modify or refer to them. • Understand how one and two-dimensional arrays function and when to integrate them accordingly when writing programs. • Program with an object-oriented paradigm in mind through the creation of user-defined methods. • Understand the existence and scope of variables and how they are affected through various actions throughout the structure of a program. • Use various commands from the math library in the context of a program. • Examine and debug compiler and runtime errors within programs.	• Short Answer Assignment • Graded Programming Assignment • Mastery Exercises • Module Exam
5 – Coding Applications	• Program with an object-oriented paradigm in mind through the creation of user-defined methods. • Use various commands from the math library in the context of a program. • Examine and debug compiler and runtime errors within programs. • Understand how content is implemented in a student-centered classroom as it applies to logic and algorithmic design.	• Graded Programming Assignment
6 – Reflection & Final Exam	• Final Exam Review	• Lab • Short Answer Assignment • Reflection Paper • Module Exam