

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 Principles uses a free digital textbook accessible at: <https://njctl.org/materials/courses/ap-computer-science-principles/>
- Participants will download SMART Notebook presentations, homework files, labs, and teacher resources from the PSI AP Computer Science Principles

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 Python 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 occurs in 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
1 – Introduction to Programming	<ul style="list-style-type: none"> • Differentiate between data types, strings and variables. • Concatenate strings. • Understand coding input and output. • Correctly use comments in Python to annotate your programs. • Debug Python code when basic errors are identified. 	<ul style="list-style-type: none"> • Short Answer Assignment • Lab • Mastery Exercises • Module Exam
2 – Operators & Logic	<ul style="list-style-type: none"> • Write comparisons in Python using math operators, including modulus. • Apply order of operations correctly, as it pertains to writing simple programs. • Write programs using comparison and logical operators. • Write programs using <code>Random()</code>. 	<ul style="list-style-type: none"> • Short Answer Assignment • Lab • Mastery Exercises • Module Exam
3 – Algorithms & Control Structure	<ul style="list-style-type: none"> • Differentiate between what is and what is not an algorithm. • Differentiate between the three different types of control structures; sequence, iteration and selection. • Write efficient programs using the most appropriate control structure as an organizational planning structure. • Write code using nesting to repeat specific portions of a program. 	<ul style="list-style-type: none"> • Short Answer Assignment • Lab • Mastery Exercise • Module Exam
4 - Lists	<ul style="list-style-type: none"> • Create lists in Python. • Traverse and search lists. • Compare strings and lists to find the commonality of each. 	<ul style="list-style-type: none"> • Short Answer Assignment • Lab • Mastery Exercise • Module Exam
5 – Functions & Procedures	<ul style="list-style-type: none"> • Utilize basic functions in programs. • Create lists using functions. • Return values and vary parameters. 	<ul style="list-style-type: none"> • Short Answer Assignment • Lab • Mastery Exercises

		<ul style="list-style-type: none">● Module Exam
6 – Reflection & Final Exam	<ul style="list-style-type: none">● Final Exam Review	<ul style="list-style-type: none">● Reflection Paper● Module Exam