

**INDIANA AREA SCHOOL DISTRICT
INDIANA, PA
Secondary**

Course Title: Computer Programming and Coding I

Grade Level: Grades 9-12

Course Number:

Core or Elective: Elective

Periods Per Week: Five

Length of Time: 39-Minutes Per Class Period

Length of Course: One semester

Units of Credit: One-half Credit

Created on: October 24, 2018

Revised: _____

Revised: _____

Revised: _____

NCAA Core Course Approved

IDEAL Course

Board Approval Date: _____

Course Description

This course is designed as a first course in computer science programming. Students will learn the basics of computer programming and computer science. The material emphasizes computational thinking and helps develop the ability to solve problems. This course covers the basic building blocks of programming along with other central elements of computer science. The concepts covered in the course include: program design and coding, number calculations and data, decision making, repetition and loops, and graphics. The primary language for the course is Python. It gives a foundation in the tools used in computer science and prepares students for further study in computer science, including the Computer Programming and Coding II and AP Computer Science course. Students taking this course must have completed Algebra II or be taking the course concurrently.

Expected Level of Achievements (District Grading Scale)

A (4):	93% - 100%
B (3):	85% - 92%
C (2):	77% - 84%
D (1):	69% - 76%
F (0):	68% and below

Academic/Content Standards/ Benchmarks

(Standards met in this course – standards specific to each unit are listed with each unit)

PDE BCIT & Mathematics Standards:

- 15.4.12.A: Apply the creative and productive use of emerging technologies for educational and personal success.
- 15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.
- 15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.
- 15.4.12.J: Create a complex computer program to solve a problem.
- 02.5.11.A: Develop a plan to analyze a problem, identify the information needed to solve the problem, carry out the plan, check whether an answer makes sense, and explain how the problem was solved in grade appropriate contexts.

PDE Computer Science Standards:

- 3A-CS-02: Compare levels of abstraction and interactions between application software, system software, and hardware layers.
- 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.
- 3A-AP-14: Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.
- 3A-AP-15: Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.
- 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
- 3A-AP-18: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
- 3A-AP-19: Systematically design and develop programs for broad audiences by incorporating feedback from users.
- 3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible.
- 3A-AP-22: Design and develop computational artifacts working in team roles using collaborative tools.
- 3A-AP-23: Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.
- 3B-AP-10: Use and adapt classic algorithms to solve computational problems.
- 3B-AP-11: Evaluate algorithms in terms of their efficiency, correctness, and clarity.
- 3B-AP-12: Compare and contrast fundamental data structures and their uses.

National Business Education Association Standards:

NBEA Computation Standards:

- I. Mathematical Foundations: Apply basic mathematical operations to solve problems.
- II. Number Relationships and Operations: Solve problems involving whole numbers, decimals, fractions, percents, ratios, averages, and proportions.
- III. Patterns, Functions, and Algebra: Use algebraic operations to solve problems.
- IV. Measurements: Use common international standards of measurement when solving problems.
- V. Statistics and Probability: Analyze and interpret data using common statistical procedures.
- VI. Problem-Solving Applications: Use mathematical procedures to analyze and solve business problems.

NBEA Information Technology Standards:

- XI. Programming and Application Development: Design, develop, test, and implement programs.
- XII. Telecommunications and Networking Infrastructures: Develop the skills to design, deploy, and administer networks and telecommunications systems.

Subject Area: Computer Science

Course Title: Computer Programming and Coding I

Grade: 9-12

Strand: Introduction to Computer Science

Time: 4 Weeks

Academic/Content Standards/ Benchmarks	Objectives	Instructional Strategies	Assessment Strategies
<p>PDE BCIT & Mathematics Standards: 15.4.12.A 15.4.12.H 15.4.12.I 15.4.12.J 2.5.11.A</p> <p>PDE Computer Science Standards: 3A-CS-02 3A-AP-14 3A-AP-17 3A-AP-21</p> <p>National Business Education Association Standards:</p> <p>NBEA Computation Standards: I, II, III, IV, V, and VI</p> <p>NBEA Information Technology Standards: XI and XII</p>	<p>The student will:</p> <ul style="list-style-type: none">▪ Define computer science and coding and list the differences between natural and formal languages.▪ Define Integrated Development Environment, open CodeSkulptor or Visual Studio, and create his/her first line of code.▪ Define output, correctly utilize the print command, and create his/her first program.▪ Define hardware, software, and programs and list the parts of the computer and types of software.▪ Define and utilize comments, escape characters, and special characters, and explain output.▪ Define and utilize input and variables, as well as describe the rules for good variable naming.▪ Define and utilize variables.▪ Distinguish between the variable types of integer and string.	<ul style="list-style-type: none">▪ Audio-visual Presentation▪ Board Work▪ Brainstorming▪ Class Discussion▪ Computer-based instruction▪ Cooperative Learning▪ Demonstrations▪ Differentiated Instruction▪ Graphic Organizer▪ Hands-on Computer Activities▪ Independent Reading/Study▪ Lecture▪ Peer Collaboration▪ Peer Tutoring▪ Performance-based Learning▪ Question and Answer▪ Role Playing▪ Teacher Modeling▪ Web-based/Online Learning	<ul style="list-style-type: none">▪ Active Daily Participation▪ Formative Assessments▪ Homework Assignments▪ Independent Study▪ Performance-based Assessments▪ Portfolio▪ Projects▪ Real world projects▪ Rubrics▪ Student Self-reflection▪ Teacher Observations▪ Teacher-made Assessments▪ Tests/Quizzes▪ Web-based Projects/Assignments

- | | | |
|---|--|--|
| <ul style="list-style-type: none">▪ Describe analog data, digital data, and data compression.▪ Describe and define computer memory and binary code.▪ Convert a number from decimal to binary representation.▪ Describe how computer science is used in almost every industry across the globe. | | |
|---|--|--|

(over)

Materials/Resources	Reteaching Strategies include:	Enrichment Enrichment options include:
<ul style="list-style-type: none"> ▪ Bulletin Boards ▪ Computers ▪ Course Resources: starter files ▪ Edhesive Introduction to CS course ▪ Python Integrated Development Environment (i.e. CodeSkulptor) ▪ Google Classroom ▪ Google Suite ▪ Internet ▪ Lab facilities ▪ Promethean Board ▪ Student and Teacher Forums (Piazza; online) ▪ Supplemental textbooks ▪ Teacher-made Materials ▪ White Boards and Markers 	<ul style="list-style-type: none"> ▪ Additional Time ▪ Alternate activity/instruction ▪ Cooperative learning activity/Peer Tutoring ▪ Graphic organizer ▪ Modified assignment ▪ Modified environment ▪ Modified expectations ▪ Note-taking practice ▪ One-on-one Instruction ▪ Online Videos/Tutorials ▪ Positive reinforcement ▪ Practice assignment ▪ Preferential Seating ▪ Study skills tutorial ▪ Supplemental reading assignment ▪ Testing modifications ▪ Verbal cue ▪ Visual cue 	<ul style="list-style-type: none"> ▪ Accelerated assignment schedule ▪ Alternate activity/instruction ▪ Computer/online software-based activity ▪ Cooperative learning activity ▪ Enrichment Assignments ▪ High Expectations ▪ Modified assignment ▪ Participation in software design competition(s) ▪ Positive reinforcement ▪ Project ▪ Seminar discussion ▪ Supplemental reading assignment

Subject Area: Computer Science

Course Title: Computer Programming and Coding I

Grade: 9-12

Strand: Number Calculations and Data

Time: 3 Weeks

Academic/Content Standards/ Benchmarks	Objectives	Instructional Strategies	Assessment Strategies
<p>PDE BCIT & Mathematics Standards: 15.4.12.A 15.4.12.H 15.4.12.I 15.4.12.J 2.5.11.A</p> <p>PDE Computer Science Standards: 3A-CS-02 3A-AP-13 3A-AP-14 3A-AP-15 3A-AP-17 3A-AP-18 3A-AP-21 3B-AP-10 3B-AP-12</p> <p>National Business Education Association Standards:</p> <p>NBEA Computation Standards: I, II, III, IV, V, and VI</p> <p>NBEA Information Technology Standards: XI and XII</p>	<p>The student will:</p> <ul style="list-style-type: none">▪ Define and utilize float, operators and assignments.▪ List the symbols for the basic operators, exponents, and the order of operations.▪ Define and utilize modular division.▪ Define and utilize functions and modules.▪ Calculate the square roots, absolute values, and floats of numbers.▪ Generate random numbers with the appropriate functions.▪ Use the random function to generate random colors.▪ Define and utilize RGB and hexadecimal format to create colors.▪ Define big data and terabytes.▪ Explain why computers are needed to compute big data.▪ Define and code max and min.▪ Define plotting data and real data set.▪ Describe the work & required skills set of data scientists and business analysts.	<ul style="list-style-type: none">▪ Audio-visual Presentation▪ Board Work▪ Brainstorming▪ Class Discussion▪ Computer-based instruction▪ Cooperative Learning▪ Demonstrations▪ Differentiated Instruction▪ Graphic Organizer▪ Hands-on Computer Activities▪ Independent Reading/Study▪ Lecture▪ Peer Collaboration▪ Peer Tutoring▪ Performance-based Learning▪ Question and Answer▪ Role Playing▪ Teacher Modeling▪ Web-based/Online Learning	<ul style="list-style-type: none">▪ Active Daily Participation▪ Formative Assessments▪ Homework Assignments▪ Independent Study▪ Performance-based Assessments▪ Portfolio▪ Projects▪ Real world projects▪ Rubrics▪ Student Self-reflection▪ Teacher Observations▪ Teacher-made Assessments▪ Tests/Quizzes▪ Web-based Projects/Assignments

(over)

Materials/Resources	Reteaching Strategies include:	Enrichment Enrichment options include:
<ul style="list-style-type: none"> ▪ Bulletin Boards ▪ Computers ▪ Course Resources: starter files ▪ Edhesive Introduction to CS course ▪ Python Integrated Development Environment (i.e. CodeSkulptor) ▪ Google Classroom ▪ Google Suite ▪ Internet ▪ Lab facilities ▪ Promethean Board ▪ Student and Teacher Forums (Piazza; online) ▪ Supplemental textbooks ▪ Teacher-made Materials ▪ White Boards and Markers 	<ul style="list-style-type: none"> ▪ Additional Time ▪ Alternate activity/instruction ▪ Cooperative learning activity/Peer Tutoring ▪ Graphic organizer ▪ Modified assignment ▪ Modified environment ▪ Modified expectations ▪ Note-taking practice ▪ One-on-one Instruction ▪ Online Videos/Tutorials ▪ Positive reinforcement ▪ Practice assignment ▪ Preferential Seating ▪ Study skills tutorial ▪ Supplemental reading assignment ▪ Testing modifications ▪ Verbal cue ▪ Visual cue 	<ul style="list-style-type: none"> ▪ Accelerated assignment schedule ▪ Alternate activity/instruction ▪ Computer/online software-based activity ▪ Cooperative learning activity ▪ Enrichment Assignments ▪ High Expectations ▪ Modified assignment ▪ Participation in software design competition(s) ▪ Positive reinforcement ▪ Project ▪ Seminar discussion ▪ Supplemental reading assignment

Subject Area: Computer Science

Course Title: Computer Programming and Coding I

Grade: 9-12

Strand: Decision Structures

Time: 3 Weeks

Academic/Content Standards/ Benchmarks	Objectives	Instructional Strategies	Assessment Strategies
<p>PDE BCIT & Mathematics Standards: 15.4.12.A 15.4.12.H 15.4.12.I 15.4.12.J 2.5.11.A</p> <p>PDE Computer Science Standards: 3A-CS-02 3A-AP-13 3A-AP-14 3A-AP-15 3A-AP-17 3A-AP-18 3A-AP-19 3A-AP-21 3A-AP-22 3A-AP-23 3B-AP-10 3B-AP-11 3B-AP-12</p> <p>National Business Education Association Standards:</p> <p>NBEA Computation Standards: I, II, III, IV, V, and VI</p> <p>NBEA Information Technology Standards: XI and XII</p>	<p>The student will:</p> <ul style="list-style-type: none">▪ Define and code parameters and explain the guidelines of functions.▪ Define and code if statements, define syntax, and list the symbols for relational operators.▪ Define and utilize Booleans (and, not, or).▪ Define and code if-else statements.▪ Define and code nested if statements.▪ Determine when it would be appropriate to use an if statement, if-else statement, or nested if statement.▪ Define, analyze, utilize, and list the rules and standards of a good algorithm.▪ Code an algorithm that uses if statements and Boolean conditions.▪ State how technology and computer science is impacting technology.	<ul style="list-style-type: none">▪ Audio-visual Presentation▪ Board Work▪ Brainstorming▪ Class Discussion▪ Computer-based instruction▪ Cooperative Learning▪ Demonstrations▪ Differentiated Instruction▪ Graphic Organizer▪ Hands-on Computer Activities▪ Independent Reading/Study▪ Lecture▪ Peer Collaboration▪ Peer Tutoring▪ Performance-based Learning▪ Question and Answer▪ Role Playing▪ Teacher Modeling▪ Web-based/Online Learning	<ul style="list-style-type: none">▪ Active Daily Participation▪ Formative Assessments▪ Homework Assignments▪ Independent Study▪ Performance-based Assessments▪ Portfolio▪ Projects▪ Real world projects▪ Rubrics▪ Student Self-reflection▪ Teacher Observations▪ Teacher-made Assessments▪ Tests/Quizzes▪ Web-based Projects/Assignments

(over)

Materials/Resources	Reteaching Strategies include:	Enrichment Enrichment options include:
<ul style="list-style-type: none"> ▪ Bulletin Boards ▪ Computers ▪ Course Resources: starter files ▪ Edhesive Introduction to CS course ▪ Python Integrated Development Environment (i.e. CodeSkulptor) ▪ Google Classroom ▪ Google Suite ▪ Internet ▪ Lab facilities ▪ Promethean Board ▪ Student and Teacher Forums (Piazza; online) ▪ Supplemental textbooks ▪ Teacher-made Materials ▪ White Boards and Markers 	<ul style="list-style-type: none"> ▪ Additional Time ▪ Alternate activity/instruction ▪ Cooperative learning activity/Peer Tutoring ▪ Graphic organizer ▪ Modified assignment ▪ Modified environment ▪ Modified expectations ▪ Note-taking practice ▪ One-on-one Instruction ▪ Online Videos/Tutorials ▪ Positive reinforcement ▪ Practice assignment ▪ Preferential Seating ▪ Study skills tutorial ▪ Supplemental reading assignment ▪ Testing modifications ▪ Verbal cue ▪ Visual cue 	<ul style="list-style-type: none"> ▪ Accelerated assignment schedule ▪ Alternate activity/instruction ▪ Computer/online software-based activity ▪ Cooperative learning activity ▪ Enrichment Assignments ▪ High Expectations ▪ Modified assignment ▪ Participation in software design competition(s) ▪ Positive reinforcement ▪ Project ▪ Seminar discussion ▪ Supplemental reading assignment

Subject Area: Computer Science

Course Title: Computer Programming and Coding I

Grade: 9-12

Strand: Repetition and Loops

Time: 3 Weeks

Academic/Content Standards/ Benchmarks	Objectives	Instructional Strategies	Assessment Strategies
<p>PDE BCIT & Mathematics Standards: 15.4.12.A 15.4.12.H 15.4.12.I 15.4.12.J 2.5.11.A</p> <p>PDE Computer Science Standards: 3A-CS-02 3A-AP-13 3A-AP-14 3A-AP-15 3A-AP-17 3A-AP-18 3A-AP-19 3A-AP-21 3A-AP-22 3A-AP-23 3B-AP-10 3B-AP-11 3B-AP-12</p> <p>National Business Education Association Standards:</p> <p>NBEA Computation Standards: I, II, III, IV, V, and VI</p> <p>NBEA Information Technology Standards: XI and XII</p>	<p>The student will:</p> <ul style="list-style-type: none">▪ Explain the use of loops in programming.▪ Define and code while loops and loop control variables.▪ Create a "while" loop to repeat a process.▪ Explain the use of counters and accumulators.▪ Create nested loops.▪ Implement algorithms using while loops.▪ Define initializing.▪ Define and code the count and sum variables.▪ End loops using count variables and user input and state when to use them.▪ Utilize the max and min functions within a loop.▪ Explain the difference between gaming for fun and gaming with a purpose.▪ State the standards games should meet to make them successful and purposeful.▪ Code a simple number game.▪ Define cybersecurity and list ways to be safe online.▪ Describe the skills required and job functions of a video game designer.	<ul style="list-style-type: none">▪ Audio-visual Presentation▪ Board Work▪ Brainstorming▪ Class Discussion▪ Computer-based instruction▪ Cooperative Learning▪ Demonstrations▪ Differentiated Instruction▪ Graphic Organizer▪ Hands-on Computer Activities▪ Independent Reading/Study▪ Lecture▪ Peer Collaboration▪ Peer Tutoring▪ Performance-based Learning▪ Question and Answer▪ Role Playing▪ Teacher Modeling▪ Web-based/Online Learning	<ul style="list-style-type: none">▪ Active Daily Participation▪ Formative Assessments▪ Homework Assignments▪ Independent Study▪ Performance-based Assessments▪ Portfolio▪ Projects▪ Real world projects▪ Rubrics▪ Student Self-reflection▪ Teacher Observations▪ Teacher-made Assessments▪ Tests/Quizzes▪ Web-based Projects/Assignments

(over)

Materials/Resources	Reteaching Strategies include:	Enrichment Enrichment options include:
<ul style="list-style-type: none"> ▪ Bulletin Boards ▪ Computers ▪ Course Resources: starter files ▪ Edhesive Introduction to CS course ▪ Python Integrated Development Environment (i.e. CodeSkulptor) ▪ Google Classroom ▪ Google Suite ▪ Internet ▪ Lab facilities ▪ Promethean Board ▪ Student and Teacher Forums (Piazza; online) ▪ Supplemental textbooks ▪ Teacher-made Materials ▪ White Boards and Markers 	<ul style="list-style-type: none"> ▪ Additional Time ▪ Alternate activity/instruction ▪ Cooperative learning activity/Peer Tutoring ▪ Graphic organizer ▪ Modified assignment ▪ Modified environment ▪ Modified expectations ▪ Note-taking practice ▪ One-on-one Instruction ▪ Online Videos/Tutorials ▪ Positive reinforcement ▪ Practice assignment ▪ Preferential Seating ▪ Study skills tutorial ▪ Supplemental reading assignment ▪ Testing modifications ▪ Verbal cue ▪ Visual cue 	<ul style="list-style-type: none"> ▪ Accelerated assignment schedule ▪ Alternate activity/instruction ▪ Computer/online software-based activity ▪ Cooperative learning activity ▪ Enrichment Assignments ▪ High Expectations ▪ Modified assignment ▪ Participation in software design competition(s) ▪ Positive reinforcement ▪ Project ▪ Seminar discussion ▪ Supplemental reading assignment

Subject Area: Computer Science

Course Title: Computer Programming and Coding I

Grade: 9-12

Strand: Graphics

Time: 5 Weeks

Academic/Content Standards/ Benchmarks	Objectives	Instructional Strategies	Assessment Strategies
<p>PDE BCIT & Mathematics Standards: 15.4.12.A 15.4.12.H 15.4.12.I 15.4.12.J 2.5.11.A</p> <p>PDE Computer Science Standards: 3A-CS-02 3A-AP-13 3A-AP-14 3A-AP-15 3A-AP-17 3A-AP-18 3A-AP-19 3A-AP-21 3A-AP-22 3A-AP-23 3B-AP-10 3B-AP-11 3B-AP-12</p> <p>National Business Education Association Standards:</p> <p>NBEA Computation Standards: I, II, III, IV, V, and VI</p> <p>NBEA Information Technology Standards: XI and XII</p>	<p>The student will:</p> <ul style="list-style-type: none">▪ Explain the use of loops in programming.▪ Create a "for" loop to repeat a process.▪ Explain the use of counters and accumulators.▪ Create nested loops.▪ Select the best type of loop to use in different situations/programs.▪ Implement algorithms using different types of loops.▪ Define and code a for loop, while loop, and range function. You should also be able to differentiate when to use a for or while loop.▪ Define and code utilizing the x and y coordinates, frames, and the def function.▪ Create drawings using the canvas.draw_line and draw polygon functions.▪ Write code to draw a house.▪ Create drawings using the draw_circle and arc functions.▪ Write code to draw emoticons.▪ Create a kaleidoscope using Google's Made with Code.▪ Define global variables and write code to animate drawings.▪ List the ways technology and computer science is impacting the film industry.▪ State how math and computer science are impacting the animation industry.	<ul style="list-style-type: none">▪ Audio-visual Presentation▪ Board Work▪ Brainstorming▪ Class Discussion▪ Computer-based instruction▪ Cooperative Learning▪ Demonstrations▪ Differentiated Instruction▪ Graphic Organizer▪ Hands-on Computer Activities▪ Independent Reading/Study▪ Lecture▪ Peer Collaboration▪ Peer Tutoring▪ Performance-based Learning▪ Question and Answer▪ Role Playing▪ Teacher Modeling▪ Web-based/Online Learning	<ul style="list-style-type: none">▪ Active Daily Participation▪ Formative Assessments▪ Homework Assignments▪ Independent Study▪ Performance-based Assessments▪ Portfolio▪ Projects▪ Real world projects▪ Rubrics▪ Student Self-reflection▪ Teacher Observations▪ Teacher-made Assessments▪ Tests/Quizzes▪ Web-based Projects/Assignments

(over)

Materials/Resources	Reteaching Strategies include:	Enrichment Enrichment options include:
<ul style="list-style-type: none"> ▪ Bulletin Boards ▪ Computers ▪ Course Resources: starter files ▪ Edhesive Introduction to CS course ▪ Python Integrated Development Environment (i.e. CodeSkulptor) ▪ Google Classroom ▪ Google Suite ▪ Internet ▪ Lab facilities ▪ Promethean Board ▪ Student and Teacher Forums (Piazza; online) ▪ Supplemental textbooks ▪ Teacher-made Materials ▪ White Boards and Markers 	<ul style="list-style-type: none"> ▪ Additional Time ▪ Alternate activity/instruction ▪ Cooperative learning activity/Peer Tutoring ▪ Graphic organizer ▪ Modified assignment ▪ Modified environment ▪ Modified expectations ▪ Note-taking practice ▪ One-on-one Instruction ▪ Online Videos/Tutorials ▪ Positive reinforcement ▪ Practice assignment ▪ Preferential Seating ▪ Study skills tutorial ▪ Supplemental reading assignment ▪ Testing modifications ▪ Verbal cue ▪ Visual cue 	<ul style="list-style-type: none"> ▪ Accelerated assignment schedule ▪ Alternate activity/instruction ▪ Computer/online software-based activity ▪ Cooperative learning activity ▪ Enrichment Assignments ▪ High Expectations ▪ Modified assignment ▪ Participation in software design competition(s) ▪ Positive reinforcement ▪ Project ▪ Seminar discussion ▪ Supplemental reading assignment