

Unit 2: Programming Copied

Content Area: **Technology**
Course(s): **Sample Course**
Time Period: **OctNov**
Length: **Sample Length & Grade Level**
Status: **Published**

AP Computer Science Principles

Department of Curriculum and Instruction



Belleville Public Schools

Curriculum Guide

AP Computer Science Principles, Grades 9-12

Unit 2, Programming

Belleville Board of Education

56 Ralph Street

Belleville, NJ 07109

Prepared by: Teacher, Corey Woodring

Dr. Richard Tomko, Ph.D., M.J., Superintendent of Schools

Ms. LucyAnn Demikoff, Director of Curriculum and Instruction K-12

Ms. Nicole Shanklin, Director of Elementary Education K-8

Mr. Joseph Lepo, Director of Secondary Education

Board Approved:

Unit Overview

Use of a programming environment to explore sequencing, selection, and iteration as part of the goal to create programs that serve useful functions.

Enduring Understanding

- Computing facilitates exploration and the creation of computational artifacts and new knowledge that help people solve personal, societal, and global problems.
- Computing allows people to use creative development processes to produce computational artifacts for creative expression or to solve a problem.
- Computing has global effects both beneficial and harmful on people and society.

Essential Questions

- How are programs developed to help people, organizations, or society solve problems?
- How does abstraction make the development of computer programs possible?

Exit Skills

Students will be able to:

Visual Programming:

- Utilize a graphical editor to read, construct, and execute dynamic programs.
- Assess, modify, and execute programs developed by others.
- Examine how well-specified behavior of objects can be constructed through sequential actions and operations.

Program State:

- Develop a variety of programs using methods and techniques that are appropriate for the goals of the programmer.
- Create programs that incorporate dynamic, user-driven, keyboard controls and input.
- Experiment with how the dynamic state of an object or program can be stored and changed using variables.
- Analyze the role of clear, descriptive names for objects, behaviors, variables, and other identifiers in maintaining the readability of code.
- Identify additional desired outcomes for a program that extend beyond its original purpose.

Selection Statements:

- Examine the uses of selection statements in programming.
- Analyze the differences between simple selection and complex, nested selection statements.
- Constructing complex conditional statements with the use of the Boolean operators "AND", "OR", and "NOT" in coding.

Coding Skills:

- Explain how algorithms are implemented using program instructions that are processed sequentially during program execution.
- Design and construct instructions using a non-traditional, domain specific notation.
- Evaluate the clarity and legibility of instructions written in a nontraditional, domain-specific notation by reading and executing instructions created by others.
- Examine a number of common programming errors.
- Explore a number of common debugging strategies.
- Develop solution and strategies for correcting common programming errors.

Global Impact:

- Examine and discuss the motivations behind a number of high profile individuals in the field of programming.
- Discuss the benefits of programming as a tool and a profession.

Analyze the legal and ethical concerns of open source and licensed software, libraries, and code.

New Jersey Student Learning Standards (NJSL-S)

CS.9-12.8.1.12.CS.1	Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
CS.9-12.8.1.12.CS.2	Model interactions between application software, system software, and hardware.
CS.9-12.8.1.12.CS.3	Compare the functions of application software, system software, and hardware.
CS.9-12.8.1.12.NI.1	Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
CS.9-12.CS	Computing Systems
CS.9-12.NI	Networks and the Internet
TECH.9.4.12.CI	Creativity and Innovation
TECH.9.4.12.CI.1	Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
TECH.9.4.12.CI.2	Identify career pathways that highlight personal talents, skills, and abilities (e.g., 1.4.12prof.CR2b, 2.2.12.LF.8).
TECH.9.4.12.CI.3	Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
TECH.9.4.12.CT	Critical Thinking and Problem-solving
TECH.9.4.12.CT.1	Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
TECH.9.4.12.CT.2	Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).
TECH.9.4.12.CT.3	Enlist input from a variety of stakeholders (e.g., community members, experts in the field) to design a service learning activity that addresses a local or global issue (e.g., environmental justice).
TECH.9.4.12.CT.4	Participate in online strategy and planning sessions for course-based, school-based, or other project and determine the strategies that contribute to effective outcomes. A computing system involves interaction among the user, hardware, application software, and system software. With a growth mindset, failure is an important part of success. Network security depends on a combination of hardware, software, and practices that protect data while it is at rest, in transit, and in use. The needs of users and the sensitivity

of data determine the level of security implemented. Advanced attacks take advantage of common security vulnerabilities.

Innovative ideas or innovation can lead to career opportunities.

The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.

Interdisciplinary Connections

LA.RL.11-12	Reading Literature Key Ideas and Details
LA.RL.11-12.1	Cite strong and thorough textual evidence and make relevant connections to support analysis of what the text says explicitly as well as inferences drawn from the text, including determining where the text leaves matters uncertain.
LA.RL.11-12.2	Determine two or more themes or central ideas of a text and analyze their development over the course of the text, including how they interact and build on one another to produce a complex account; provide an objective summary of the text.
LA.RL.11-12.3	Analyze the impact of the author's choices regarding how to develop and relate elements of a story or drama (e.g., where a story is set, how the action is ordered, how the characters are introduced and developed).
LA.L.11-12.4	Determine or clarify the meaning of unknown and multiple-meaning words and phrases based on grades 11–12 reading and content, choosing flexibly from a range of strategies.
LA.L.11-12.4.A	Use context (e.g., the overall meaning of a sentence, paragraph, or text; a word's position or function in a sentence) as a clue to the meaning of a word or phrase.
LA.L.11-12.4.C	Consult general and specialized reference materials (e.g., dictionaries, glossaries, thesauruses), both print and digital, to find the pronunciation of a word or determine or clarify its precise meaning, its part of speech, its etymology, or its standard usage.
LA.L.11-12.4.D	Verify the preliminary determination of the meaning of a word or phrase (e.g., by checking the inferred meaning in context or in a dictionary).

Learning Objectives

Develop a variety of programs using methods and techniques that are appropriate for the goals of the programmer.

- Create programs that incorporate dynamic, user-driven, keyboard controls and input.
- Experiment with how the dynamic state of an object or program can be stored and changed using variables.
- Analyze the role of clear, descriptive names for objects, behaviors, variables, and other identifiers in maintaining the readability of code.
- Identify additional desired outcomes for a program that extend beyond its original purpose.

Suggested Activities & Best Practices

- Student will be able to identify famous African Americans in the field of computer programming
- Students will also be introduced to the basic energy efficient models in the programming field to help reduce global warming

Visual Programming:

- Utilize a graphical editor to read, construct, and execute dynamic programs.
- Assess, modify, and execute programs developed by others.
- Examine how well-specified behavior of objects can be constructed through sequential actions and operations.

Program State:

Develop a variety of programs using methods and techniques that are appropriate for the goals of the programmer.

- Create programs that incorporate dynamic, user-driven, keyboard controls and input.
- Experiment with how the dynamic state of an object or program can be stored and changed using variables.
- Analyze the role of clear, descriptive names for objects, behaviors, variables, and other identifiers in maintaining the readability of code.
- Identify additional desired outcomes for a program that extend beyond its original purpose.

Selection Statements:

- Examine the uses of selection statements in programming.
- Analyze the differences between simple selection and complex, nested selection statements.
- Constructing complex conditional statements with the use of the Boolean operators "AND", "OR", and "NOT" in coding.

Coding Skills:

- Explain how algorithms are implemented using program instructions that are processed sequentially during program execution.
- Design and construct instructions using a non-traditional, domain specific notation.
- Evaluate the clarity and legibility of instructions written in a nontraditional, domain-specific notation by reading and executing instructions created by others.
- Examine a number of common programming errors.
- Explore a number of common debugging strategies.
- Develop solution and strategies for correcting common programming errors.

Global Impact:

- Examine and discuss the motivations behind a number of high profile individuals in the field of programming.
- Discuss the benefits of programming as a tool and a profession.
- Analyze the legal and ethical concerns of open source and licensed software, libraries, and code.

Assessment Evidence - Checking for Understanding (CFU)

Formative Assessments

- Think, pair, share review questions from text.
- Practice mini-programs to strengthen concepts as taught.
- Teacher Observation
- Utilizing Gliffy.com to flowchart programs

Summative Assessments

- Chapter Test
- End of Chapter Projects from book.

example: <https://quizlet.com/76598385/ap-computer-science-a-flash-cards/>

Alternate Assessment

Written reports

Benchmark Assessments

- Design and construct instructions using a non-traditional, domain specific notation.
- Evaluate the clarity and legibility of instructions written in a nontraditional, domain-specific notation

by reading and executing instructions created by others.

- Admit Tickets
- Anticipation Guide
- Common Benchmarks
- Compare & Contrast
- Create a Multimedia Poster
- DBQ's
- Define
- Describe
- Evaluate
- Evaluation rubrics
- Exit Tickets
- Explaining
- Fist- to-Five or Thumb-Ometer
- Illustration
- Journals
- KWL Chart
- Learning Center Activities
- Multimedia Reports
- Newspaper Headline
- Outline
- Question Stems
- Quickwrite
- Quizzes
- Red Light, Green Light
- Self- assessments
- Socratic Seminar
- Study Guide
- Surveys
- Teacher Observation Checklist
- Think, Pair, Share
- Think, Write, Pair, Share
- Top 10 List
- Unit review/Test prep
- Unit tests

- Web-Based Assessments
- Written Reports

Primary Resources & Materials

Primary Resources:

Edhesive Online Training
Code.Org Online Training

Computers and Internet Access
AP Central at Collegeboard.org
Massive Open Online Course
Code.org

Multimedia Applications Tools

Abelson, H., Ledeen, K., and Lewis, H. R. Blown to Bits: your life, liberty, and happiness after the digital explosion. Upper Saddle River, N.J.: Addison-Wesley, 2008.

Ancillary Resources

AP Approved Programming Resources:

Alice - This 3-D modeling environment allows students to create and animate 3-D worlds. This environment lends itself well to creating stories and games.

App Inventor - This open-source Web application allows students to create their own applications on mobile devices. App Lab - This is a programming environment for creating web applications with JavaScript. It allows students to develop programs and toggle back and forth between block-based and text-based programming modes.

EarSketch - This browser-based application allows students to create their own music using either JavaScript or Python. Greenfoot - This Java IDE is designed for use in education to create two-dimensional graphic applications, such as simulations and interactive games.

Java - There are several IDEs that can be used to write in Java. The Java language allows students to create and solve problems that vary widely in difficulty.

JavaScript - This language is commonly used to create interactive effects within Web browsers.

Lego Mindstorms NXT - This product integrates programming with Lego bricks and sensors to create and program robots. The instructions are assembled by linking together function blocks.

Processing - This programming language was initially created to serve as a software sketchbook, and it can be used to teach programming using a visual context.

Python - This language has the benefit of readability that might be helpful to new programmers.

Scratch - This blocks-based programming language allows students to build scripts to run animations. This product can be downloaded and installed on a computer or run in the browser.

Snap! - This Scratch-style programming language is block-based and allows users to define new primitives in

JavaScript. Users can read and write information from the Internet using server-defined APIs and make mobile applications.

Swift - This programming language is designed for use with iOS, OS X, tvOS and watchOS. This environment allows students to create their own Apple apps and includes interactive environments that allow students to see the effects of changes or additions to code as they type.

Design and Development Process:

“What Is the Software Development Life Cycle?” Official Blog Airbrake Bug Tracker.
<https://airbrake.io/blog/insight/what-is-the-software-development-life-cycle>

“Engineering Design Process.” [https://www.teachengineering.org/ engrdesignprocess.php](https://www.teachengineering.org/engrdesignprocess.php)

“The Engineering Design Process.” <http://www.eie.org/overview/engineeringdesign-process> Mohammed, Nabil, Ali Munassar, and A. Govardhan.

“A Comparison Between Five Models of Software Engineering.” IJCSI International Journal of Computer Science 7.5 (2010): 94-101.

Open Source:

“What Is Open Source?” Opensource.com. <https://opensource.com/resources/whatsopen-source>

Open Source Initiative. <http://opensource.org/>

Technology Infusion

Please reference video links and websites listed under Ancillary Resources and Suggested Activities & Best Practices.

Technology Infusion and/or strategies include chromebooks online materials google/powerpoint slides

Alignment to 21st Century Skills & Technology

WRK.9.2.12.CAP	Career Awareness and Planning
WRK.9.2.12.CAP.1	Analyze unemployment rates for workers with different levels of education and how the economic, social, and political conditions of a time period are affected by a recession.
WRK.9.2.12.CAP.2	Develop college and career readiness skills by participating in opportunities such as structured learning experiences, apprenticeships, and dual enrollment programs.
WRK.9.2.12.CAP.3	Investigate how continuing education contributes to one's career and personal growth.
WRK.9.2.12.CAP.4	Evaluate different careers and develop various plans (e.g., costs of public, private, training schools) and timetables for achieving them, including educational/training requirements, costs, loans, and debt repayment.
TECH.9.4.12.CI	Creativity and Innovation
TECH.9.4.12.CI.1	Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
TECH.9.4.12.CI.2	Identify career pathways that highlight personal talents, skills, and abilities (e.g., 1.4.12prof.CR2b, 2.2.12.LF.8).
TECH.9.4.12.CI.3	Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
TECH.9.4.12.CT	Critical Thinking and Problem-solving
TECH.9.4.12.CT.1	Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
TECH.9.4.12.CT.2	Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).
TECH.9.4.12.CT.3	Enlist input from a variety of stakeholders (e.g., community members, experts in the field) to design a service learning activity that addresses a local or global issue (e.g., environmental justice). Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed. There are strategies to improve one's professional value and marketability. Innovative ideas or innovation can lead to career opportunities. With a growth mindset, failure is an important part of success.

21st Century Skills/Interdisciplinary Themes

21st century themes: The unit will integrate the 21st Century Life and career standard 9.1 strands A-D. These strands include: critical thinking and problem solving, creativity and innovation, collaboration, teamwork, and leadership, and cross cultural understanding and interpersonal communication

Mastery and infusion of 21st Century Skills & Technology and their Alignment to the core content areas is essential to student learning. The core content areas include:

- English Language Arts;

- Mathematics;
- Science and Scientific Inquiry (Next Generation);
- Social Studies, including American History, World History, Geography, Government and Civics, and Economics;
- World languages;
- Technology;
- Visual and Performing Arts.

- Communication and Collaboration
- Creativity and Innovation
- Critical thinking and Problem Solving
- ICT (Information, Communications and Technology) Literacy
- Information Literacy
- Life and Career Skills
- Media Literacy

21st Century Skills

21st Century Skills that will be incorporated into this unit.

- Civic Literacy
- Environmental Literacy
- Financial, Economic, Business and Entrepreneurial Literacy
- Global Awareness
- Health Literacy

CRP.K-12.CRP1.1	Career-ready individuals understand the obligations and responsibilities of being a member of a community, and they demonstrate this understanding every day through their interactions with others. They are conscientious of the impacts of their decisions on others and the environment around them. They think about the near-term and long-term consequences of their actions and seek to act in ways that contribute to the betterment of their teams, families, community and workplace. They are reliable and consistent in going beyond the minimum expectation and in participating in activities that serve the greater good.
CRP.K-12.CRP2.1	Career-ready individuals readily access and use the knowledge and skills acquired through experience and education to be more productive. They make connections between abstract concepts with real-world applications, and they make correct insights about when it is appropriate to apply the use of an academic skill in a workplace situation.
CRP.K-12.CRP4.1	Career-ready individuals communicate thoughts, ideas, and action plans with clarity, whether using written, verbal, and/or visual methods. They communicate in the workplace with clarity and purpose to make maximum use of their own and others' time. They are excellent writers; they master conventions, word choice, and organization, and use effective tone and presentation skills to articulate ideas. They are skilled at interacting with others; they are active listeners and speak clearly and with purpose. Career-ready individuals think about the audience for their communication and prepare accordingly to ensure the desired outcome.

CRP.K-12.CRP7.1

Career-ready individuals are discerning in accepting and using new information to make decisions, change practices or inform strategies. They use reliable research process to search for new information. They evaluate the validity of sources when considering the use and adoption of external information or practices in their workplace situation.

CRP.K-12.CRP8.1

Career-ready individuals readily recognize problems in the workplace, understand the nature of the problem, and devise effective plans to solve the problem. They are aware of problems when they occur and take action quickly to address the problem; they thoughtfully investigate the root cause of the problem prior to introducing solutions. They carefully consider the options to solve the problem. Once a solution is agreed upon, they follow through to ensure the problem is solved, whether through their own actions or the actions of others.

Differentiation

• Technology Resources • Teacher Tutoring • Peer Tutoring • Cooperative Learning Groups • Differentiated Instruction • Follow all IEP Modifications/504 Plan

Exemplar: One technique I use is when I teach a new Computer Science skill is using 'task cards'. For example, if we are doing a lesson on loops and there is clearly a pair of students in the class that has mastered the skill, I then have task cards with challenges ready on different skills or a harder looping concept.

Differentiations:

- Small group instruction
- Small group assignments
- Extra time to complete assignments
- Pairing oral instruction with visuals
- Repeat directions
- Use manipulatives
- Center-based instruction
- Token economy
- Study guides
- Teacher reads assessments allowed
- Scheduled breaks
- Rephrase written directions
- Multisensory approaches
- Additional time
- Preview vocabulary
- Preview content & concepts
- Story guides
- Behavior management plan
- Highlight text
- Student(s) work with assigned partner
- Visual presentation
- Assistive technology
- Auditory presentations
- Large print edition
- Dictation to scribe

- Small group setting

Hi-Prep Differentiations:

- Alternative formative and summative assessments
- Choice boards
- Games and tournaments
- Group investigations
- Guided Reading
- Independent research and projects
- Interest groups
- Learning contracts
- Leveled rubrics
- Literature circles
- Multiple intelligence options
- Multiple texts
- Personal agendas
- Project-based learning
- Problem-based learning
- Stations/centers
- Think-Tac-Toes
- Tiered activities/assignments
- Tiered products
- Varying organizers for instructions

Lo-Prep Differentiations

- Choice of books or activities
- Cubing activities
- Exploration by interest
- Flexible grouping
- Goal setting with students
- Jigsaw
- Mini workshops to re-teach or extend skills
- Open-ended activities
- Think-Pair-Share
- Reading buddies
- Varied journal prompts
- Varied supplemental materials

Special Education Learning (IEP's & 504's)

Special Education Learning adaptations that will be employed in the unit, using the ones identified below.

Exemplar: Adapting existing materials, simplifying or supplementing materials for Special Education Learning other

options are below.

- **Adjust** the method of presentation or content.
- **Develop** supplemental material.
- **Tape-record** directions for the material.
- **Provide** alternatives for responding to questions.
- **Rewrite** brief sections to lower the reading level.
- **Outline** the material for the student before reading a selection.
- **Reduce** the number of pages or items on a page to be completed by the student.
- **Break** tasks into smaller subtasks.
- **Provide** additional practice to ensure mastery.
- **Substitute** a similar, less complex task for a particular assignment.
- **Develop** simple study guides to complement required materials.

- printed copy of board work/notes provided
- additional time for skill mastery
- assistive technology
- behavior management plan
- Center-Based Instruction
- check work frequently for understanding
- computer or electronic device utilizes
- extended time on tests/ quizzes
- have student repeat directions to check for understanding
- highlighted text visual presentation
- modified assignment format
- modified test content
- modified test format
- modified test length
- multi-sensory presentation
- multiple test sessions
- preferential seating
- preview of content, concepts, and vocabulary
- Provide modifications as dictated in the student's IEP/504 plan
- reduced/shortened reading assignments
- Reduced/shortened written assignments
- secure attention before giving instruction/directions
- shortened assignments

- student working with an assigned partner
- teacher initiated weekly assignment sheet
- Use open book, study guides, test prototypes

English Language Learning (ELL)

English Language Learning adaptations that will be employed in the unit, using the ones identified below.

Exemplar: Advance Notes

One way that we can make things easier is by preparing and distributing advance notes. This gives ELLs the opportunity to preview what will be taught and, in turn, aids in comprehension of the material.

- teaching key aspects of a topic. Eliminate nonessential information
- using videos, illustrations, pictures, and drawings to explain or clarify
- allowing products (projects, timelines, demonstrations, models, drawings, dioramas, poster boards, charts, graphs, slide shows, videos, etc.) to demonstrate student's learning;
- allowing students to correct errors (looking for understanding)
- allowing the use of note cards or open-book during testing
- decreasing the amount of work presented or required
- having peers take notes or providing a copy of the teacher's notes
- modifying tests to reflect selected objectives
- providing study guides
- reducing or omitting lengthy outside reading assignments
- reducing the number of answer choices on a multiple choice test
- tutoring by peers
- using computer word processing spell check and grammar check features
- using true/false, matching, or fill in the blank tests in lieu of essay tests

At Risk

Intervention Strategies that will be employed in the unit

Exemplar: Parental Involvement

Planned intervention means involving parents. Do you have an agenda in place that goes home each night? Are parents also signing the agenda or contracts you have set up? How are you involving [parental support](#) at home for homework or additional follow up?

- allowing students to correct errors (looking for understanding)
- teaching key aspects of a topic. Eliminate nonessential information
- allowing products (projects, timelines, demonstrations, models, drawings, dioramas, poster boards, charts, graphs, slide shows, videos, etc.) to demonstrate student's learning
- allowing students to select from given choices
- allowing the use of note cards or open-book during testing
- collaborating (general education teacher and specialist) to modify vocabulary, omit or modify items to reflect objectives for the student, eliminate sections of the test, and determine how the grade will be determined prior to giving the test.
- decreasing the amount of work presented or required
- having peers take notes or providing a copy of the teacher's notes
- marking students' correct and acceptable work, not the mistakes
- modifying tests to reflect selected objectives
- providing study guides
- reducing or omitting lengthy outside reading assignments
- reducing the number of answer choices on a multiple choice test
- tutoring by peers
- using authentic assessments with real-life problem-solving
- using true/false, matching, or fill in the blank tests in lieu of essay tests
- using videos, illustrations, pictures, and drawings to explain or clarify

Talented and Gifted Learning (T&G)

Exemplar : Provide students with problem-based learning activity using multiple standards from the unit.

http://www.grandviewlibrary.org/CurriculumAdaptations/General_Gifted.pdf

Grouping • Group gifted students with other gifted students or higher-level learners. • Refrain from grouping gifted students with lower-level students for remediation.

Talented and Gifted adaptations that will be employed in the unit, using the ones identified below.

- Above grade level placement option for qualified students
- Advanced problem-solving
- Allow students to work at a faster pace
- Cluster grouping
- Complete activities aligned with above grade level text using Benchmark results
- Create a blog or social media page about their unit
- Create a plan to solve an issue presented in the class or in a text
- Debate issues with research to support arguments

- Flexible skill grouping within a class or across grade level for rigor
- Higher order, critical & creative thinking skills, and discovery
- Multi-disciplinary unit and/or project
- Teacher-selected instructional strategies that are focused to provide challenge, engagement, and growth opportunities
- Utilize exploratory connections to higher-grade concepts
- Utilize project-based learning for greater depth of knowledge

Sample Lesson

<https://docs.google.com/document/d/14kRjzsHFLi1n6fmALzZVMroOWd245ouP/edit?usp=sharing&oid=117308739238991367976&rtpof=true&sd=true>

Lesson Plan – Introduction to Programming

Teacher:	Corey Woodring	Time Frame:	11 days
Grade:	9-12	School:	Belleville High School.
Subject:	AP Computer Science Principles		

AP Essential Knowledge

(Referenced from CollegeBoard AP CS P Course & Exam Description)

AAP-1

- A. Represent a value with a variable.
 1. A variable is an abstraction inside a program that can hold a value. Each variable has associated data storage that represents one value at a time, but that value can be a list or other collection that in turn contains multiple values.
 2. Using meaningful variable names helps with the readability of program code and understanding of what values are represented by the variables.
 3. Some programming languages provide types to represent data, which are referenced using variables. These types include numbers, Booleans, lists, and strings.
 4. Some values are better suited to representation using one type of datum rather than another.
- A. Determine the value of a variable as a result of an assignment.
 1. The assignment operator allows a program to change the value represented by a variable.
 2. The exam reference sheet provides the "←" operator to use for assignment. For example,

Text:

`a ← expression`

Block:

`a ← expression`

evaluates expression and then assigns a copy of the result to the variable a.

1. The value stored in a variable will be the most recent value assigned. For example:

`a ← 1`

`b ← a`

`a ← 2`

`display(b)`

still displays 1.

AAP-3

A. For procedure calls:

a. Write statements to call procedures.

b. Determine the result or effect of a procedure call.

1. The exam reference sheet provides the procedure

Text:

`DISPLAY(expression)`

Block:

`DISPLAY expression`

to display the value of expression, followed by a space.

1. The exam reference sheet provides the

Text:

`RETURN(expression)`

Block:

`RETURN expression`

statement, which is used to return the flow of control to the point where the procedure was called and to return the value of expression.

1. The exam reference sheet provides procedure

Text:

INPUT()

Block:

INPUT

which accepts a value from the user and returns the input value.

Enduring Understanding & CTP Skills

(Referenced from CollegeBoard AP CS P Course & Exam Description)

AAP-1 To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways.

AAP-3 Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing them to write programs more quickly and with more confidence.

1. Computational Solution Design

- A. Investigate the situation, context or task.
- B. Determine and design an appropriate method or approach to achieve the purpose.

2. Algorithms & Program Development

- A. Represent algorithmic processes without using a programming language.
- B. Implement an algorithm in a program.

1. Code Analysis

- A. Explain how a code segment or program functions.
- B. Determine the result of code segments.
- C. Identify and correct errors in algorithms and programs including error discovery through testing.

Essential Questions

(Some referenced from CollegeBoard AP CS P

Course & Exam Description)

(What questions will the student be able to answer as a result of the instruction?)

1. How can we store data in the Python programming language and AP Pseudocode?
2. How can we gather input and provide output in the Python programming language and AP Pseudocode?
3. How can we debug Python programs?
4. How can we describe the Python programs in plain English?

Assessment

(What is acceptable evidence to show desired results (rubrics, exam, etc.)? Attach Copy

- Complete various written checkpoint exercises that focus on the explanation and description of computer programming, pseudocode, and python.
- Develop a visual representation of the communication processes within a computer using appropriate terminology.
- Properly document a program using correct indentation, spacing, and comment style.
- Debug programs and determine the types of errors in the program.
- Create programs based on programming exercises that display various types of output using string and numeric expressions.
- Unit 1 Assessment

Formative Evaluations:

Formative Assessment with polling

codeIt! Nows

Classwork/Homework

Quizzes

AP Classroom Big Idea 1 and Big Idea 3 Formative Topic Questions (see Sequence and Scope for when to assign problems)

Long Programs (LP)/ Lab work

Sequence and Scope

Day

Topic/Activities

Summative Evaluations:

Unit 1 Test/ReTest

CW-HW

1

- Intro to Programming

Problems #1 - 4

2	<ul style="list-style-type: none"> • Output 	Problems #5 - 12
3	<ul style="list-style-type: none"> • Data Types, Strings, & Variables 	Problems #13 – 27, AP Classroom Topic Questions 3.1, 3.3
4	<ul style="list-style-type: none"> • Quiz #1 • String Concatenation 	Problems #28 - 29
5	<ul style="list-style-type: none"> • Finish String Concatenation • codeIt! Now 	Problems #30 – 31, AP Classroom Topic Questions 3.4
6	<ul style="list-style-type: none"> • Input • codeIt! Now 	Problems #32 - 35
7	<ul style="list-style-type: none"> • Quiz #2 • Comments in Python 	None
8	<ul style="list-style-type: none"> • codeIt! Now 	Problems #36 – 38
9	<ul style="list-style-type: none"> • Debugging Python • codeIt! Now 	Problems #39 – 42, AP Classroom Topic Questions 1.1
10	<ul style="list-style-type: none"> • Lab 	Study for Test
11	<ul style="list-style-type: none"> • Unit 1 Assessment 	None

CS.9-12.8.1.12.CS.1	Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
CS.9-12.8.1.12.CS.2	Model interactions between application software, system software, and hardware.
CS.9-12.8.1.12.CS.3	Compare the functions of application software, system software, and hardware.
CS.9-12.8.1.12.NI.1	Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
CS.9-12.CS	Computing Systems
CS.9-12.NI	Networks and the Internet
TECH.9.4.12.CI	Creativity and Innovation
TECH.9.4.12.CI.1	Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).

TECH.9.4.12.CI.2	Identify career pathways that highlight personal talents, skills, and abilities (e.g., 1.4.12prof.CR2b, 2.2.12.LF.8).
TECH.9.4.12.CI.3	Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
TECH.9.4.12.CT	Critical Thinking and Problem-solving
TECH.9.4.12.CT.1	Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
TECH.9.4.12.CT.2	Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).
TECH.9.4.12.CT.3	Enlist input from a variety of stakeholders (e.g., community members, experts in the field) to design a service learning activity that addresses a local or global issue (e.g., environmental justice).
TECH.9.4.12.CT.4	<p>Participate in online strategy and planning sessions for course-based, school-based, or other project and determine the strategies that contribute to effective outcomes.</p> <p>The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.</p> <p>The scalability and reliability of the Internet are enabled by the hierarchy and redundancy in networks. Network topology is determined by many characteristics.</p> <p>With a growth mindset, failure is an important part of success.</p>