Unit 1 Fundamentals of Programming Copied from: AP Computer Science A, Copied on: 02/21/22

Content Area: **CTE**
Course(s): **AP Computer Science A**
Time Period: **September**
Length: **14 days**
Status: **Published**

Fundamentals of Programming

Department of Curriculum and Instruction

Belleville Public Schools

Curriculum Guide

AP Computer Science A - Java

Fundamentals of Programming

Belleville Board of Education

102 Passaic Avenue

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, ESL Coordinator K-12

Mr. George Droste, Director of Secondary Education

Board Approved:

Unit Overview

STUDENTS WILL KNOW:

- Computer basics and hardware
- How to express and evaluate numeric expressions using basic and exponential operators
- How to name and identify variables and constants
- How to input data and assign it to a variable accordingly.
- How to name and use strings

In this unit students will edit, compile, and execute their first Java programs.

They will be able to discuss the importance of the Java programming language as well as the aspects that make it unique and powerful.

Enduring Understanding

STUDENTS WILL UNDERSTAND THAT:

- Successful creation of a program is a result of many factors in a computer system working together
- There are many ways to write and evaluate a numeric expression to perform a simple computation and that the data for an expression must be stored in the correct data type
- Students will understand that… • Java differs from other OOP languages in that its compiler translates code into java byte code for interpreting by the JVM. • Whether you use an IDE or the terminal, the edit, compile, execute process is
  how computer programs are implemented.

Essential Questions

- How are the fundamentals of a Java program and computer basics essential for successful compilation and output?
- How is a program stored and run through the hardware of a computer?
- What are the differences between Java errors and how are they debugged?
- How can Java programs be designed and written to perform simple computations that includes basic Java identifiers?
- How are variables and constants identified and named in a Java program?
- What are the primitive Java data types and what types of information do they store?
- How are operators used to evaluate numeric expressions?
- How can the value of one data type be cast onto another data type?
- What is string concatenation and how is it used?
- How can a program be designed and written to solve financial questions?
- • Why Java?
-
    - • What is the JVM and byte code?
    - • What is the edit, compile, execute procedure?
    - • What is the structure of a simple Java program?
    - • Why is design (in the form of flowcharting) so important?

Exit Skills

Assessment

- Complete various written checkpoint exercises that focus on the explanation and description of computer basics.
- Develop a visual representation of the communication processes within a computer using appropriate ter
- 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 n
- Unit 1 Assessment

Formative Evaluations:	**Summati**
Formative Assessment with polling Quizzes Components of AP approved Lab #1	Unit 1 Te

Sequence and Scope

Day	*Topic/Activities*	
1	- **Introduction** - Install an IDE/JDK (if not done already) - **Hello, World**	
2	- **Basic Syntax & Output** - CW-HW #5 - 8 - codeIt! Now	F
3	- CW-HW #10 - *Quiz Question #1*	
4	- **Escape Characters** - **Comments** - codeIt! Now	F I
5	- **Primitive Data Types**	I

	- codeIt! Now - **Variables**	
6	- Finish **Variables** - codeIt! Now	I
7	- *Quiz Question #2* - **Syntax Errors in a Program**	Fi
8	- **Arithmetic Operators** - codeIt! Now	I
9	- **Type Casting** - codeIt! Now - CW-HW #28 - 39	Fini
10	- **Input** - codeIt! Now - CW-HW #40 - 41	I
11	- *Quiz Question #3*	
12	- **The string Type** - codeIt! Now	I
13	- CW-HW #47 – 48 - AP Lab Work (if necessary)	
14	- Unit 1 Assessment	

New Jersey Student Learning Standards (NJSLS-S)

CSTA Standards
Computational Thinking • CT.L3A-07: Describe how various types of data are stored in a computer system • CT.L3B-05: Use data analysis to enhance understanding of complex natural and human systems.

Collaboration

- CL.L3A-02: Use collaborative tools to communicate with project team members (e.g. discussion threads control, etc.).
- CL.L3B-01: Use project collaboration tools, version control systems, and Integrated Development Envir working on a collaborative software project.
- CL.L3B-03: Evaluate programs written by others for readability and usability.

Computing Practice and Programming

- CPP.L3A-03: Use various debugging and testing methods to ensure program correctness (e.g., use one o models).
- CPP.L3A-04: Apply analysis, design, and implementation techniques to solve problems (e.g., use one or models).
- CPP.L3A-05: Use Application Program Interfaces (APIs) and libraries to facilitate programming solutio
- CPP.L3A-06: Select appropriate file formats for various types and uses of data.
- CPP.L3A-08: Explain the program execution process.
- CPP.L3B-01: Use advanced tools to create digital artifacts (e.g., web design, animation, video, multimed
- CPP.L3B-08: Deploy various data collection techniqes for different types of problems.

Computers and Communication Devices

- CD.L3A-01: Describe the unique features of computers embedded in mobile devices and vehicles (e.g., airplanes).
- CD.L3A-02: Develop criteria for purchasing or upgrading computer system hardware.
- CD.L3A-03: Describe the principal components of computer organization (e.g., input, output, processing
- CD.L3A-04: Compare various forms of input and output.
- CD.L3A-05: Explain the multiple levels of hardware and software that support program execution (e.g. operating systems, networks).
- CD.L3A-06: Apply strategies for identifying and solving routine hardware and software problems that o
- CD.L3A-07: Compare and contrast client-server and peer-to-peer network strategies.
- CD.L3A-08: Explain the basic components of computer networks (e.g., servers, file protection, routing, shared resources, and fault-tolerance)
- CD.L3B-04: Describe the issues that impact network functionality (e.g., latency, bandwidth, firewalls, se

TECH.8.2.12.E.3	Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications, and games).
TECH.8.2.12.E.4	Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
TECH.8.2.12.E.CS1	Computational thinking and computer programming as tools used in design and engineering.

Interdisciplinary Connections

Primary interdisciplinary connections:

Infused within the unit are connections to the 2009 NJCCCS for Language Arts Literacy and Business, Science and Technology.

Critical reading, writing, and mathematical modeling skills are promoted within the problem solving process and as a means to explain solutions.

LA.RST.11-12	Reading Science and Technical Subjects
LA.RST.11-12.1	Accurately cite strong and thorough evidence from the text to support analysis of science and technical texts, attending to precise details for explanations or descriptions.
LA.RST.11-12.5	Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
LA.RST.11-12.7	Integrate and evaluate multiple sources of information presented in diverse formats and media (e.g., quantitative data, video, multimedia) in order to address a question or solve a problem.
LA.RST.11-12.9	Synthesize information from a range of sources (e.g., texts, experiments, simulations) into a coherent understanding of a process, phenomenon, or concept, resolving conflicting information when possible.
LA.WHST.11-12.2.B	Develop the topic thoroughly by selecting the most significant and relevant facts, extended definitions, concrete details, quotations, or other information and examples appropriate to the audience's knowledge of the topic.
LA.WHST.11-12.6	Use technology, including the Internet, to produce, share, and update writing products in response to ongoing feedback, including new arguments or information.
	Determining an output value for a particular input involves evaluating an expression; finding inputs that yield a given output involves solving an equation. Questions about when two functions have the same value for the same input lead to equations, whose solutions can be visualized from the intersection of their graphs. Because functions describe relationships between quantities, they are frequently used in modeling. Sometimes functions are defined by a recursive process, which can be displayed effectively using a spreadsheet or other technology.
	Functions

Learning Objectives

A: STUDENTS WILL KNOW:

- Computer basics and hardware
- How to express and evaluate numeric expressions using basic and exponential operators
- How to name and identify variables and constants
- How to input data and assign it to a variable accordingly.
- How to name and use strings

B: STUDENTS WILL UNDERSTAND THAT:

- Successful creation of a program is a result of many factors in a computer system working together
- There are many ways to write and evaluate a numeric expression to perform a simple computation and that the data for an expression must be stored in the correct data type

C: STUDENTS WILL BE ABLE TO:

- Understand computer basics, programs and operating systems
- Create and compile a simple Java program to display output on a console
- Explain the differences between syntax errors, runtime errors, 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
- Input data.
- Perform operations using basic and exponential operators

Perform basic string concatenation

Suggested Activities & Best Practices

Assignment Ideas/Suggested Activities

- **CS Unplugged**
- **CS Fundamentals Unplugged** from Code.org
- **CS4K12** Collection of handouts, worksheets and other documents to help teach Computer Science and Engineering concepts. Materials primarily target middle-school students (6th - 8th grade), but most can be used without modification for younger or older students.
- **Code.org YouTube Channel** > Learn about computer science
- **Nifty Assignments** Collection of fun, inspirational, or thought-provoking assignments for CS1 and

CS2 shared at the annual SIGCSE conference

- **EngageCSEdu** Collection of activities and programming assignments designed to engage ALL students in CS1- and CS2-level courses. Materials have been reviewed by computer science educators and experts in student engagement and pedagogy.
- **Websheets** A system for creating and solving programming assignments **About Websheets**
- **Stanford CS Education Library** Education CS material from Stanford CS classes
- **Learn Java in N Games** Collection of game-based activities (including POGIL-style activities) for learning about Java programming. Can be used individually or as a complete curriculum. CS2 level.

Best Practices

- **CS Teaching Tips** Tips to help teachers anticipate students' difficulties and build upon students' strengths. Funded by a National Science Foundation Grant.

Assessment Evidence - Checking for Understanding (CFU)

Formative Assessments

• Think, pair, share review questions from text.

Pair Programming

• Teacher Observation

• Chapter Test

• End of Chapter Projects from Program, CS Awesome

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

-Unit test-summative asessment

Written reports-alternate assessment

Create a Multimedia poster-benchmark assessment

- 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

NJCTL Curriculum, https://njctl.org/materials/courses/ap-computer-science-a/

Curriculum development Resources/Instructional Materials/Equipment Needed Teacher Resources: • www.gliffy.com • Eclipse IDE • MS DOS Prompt • Computers

Ancillary Resources

Java Resources

- Java Review for the AP CS A Exam - Great review site with lots of practice questions.
- Aplus Compter Science Exam Review Material -Slide, Free Response, and more!
- Introduction to Java - a textbook for a first course in computer science for the next generation of scientists and engineers
- Guru-99 Introduction Java Material
- Oracles (owners of Java) has their own tutorials
- Dick Baldwin - ACC - Introduction and Advanced Java Material
- Introduction to Computer Science using Java - by Bradley Kjell
- Thinking in Java
- Blue Pelican Java
- Java Coding Bat - Lots of good practice problems
- Code Academy - No Java but good practice.

Technology Infusion

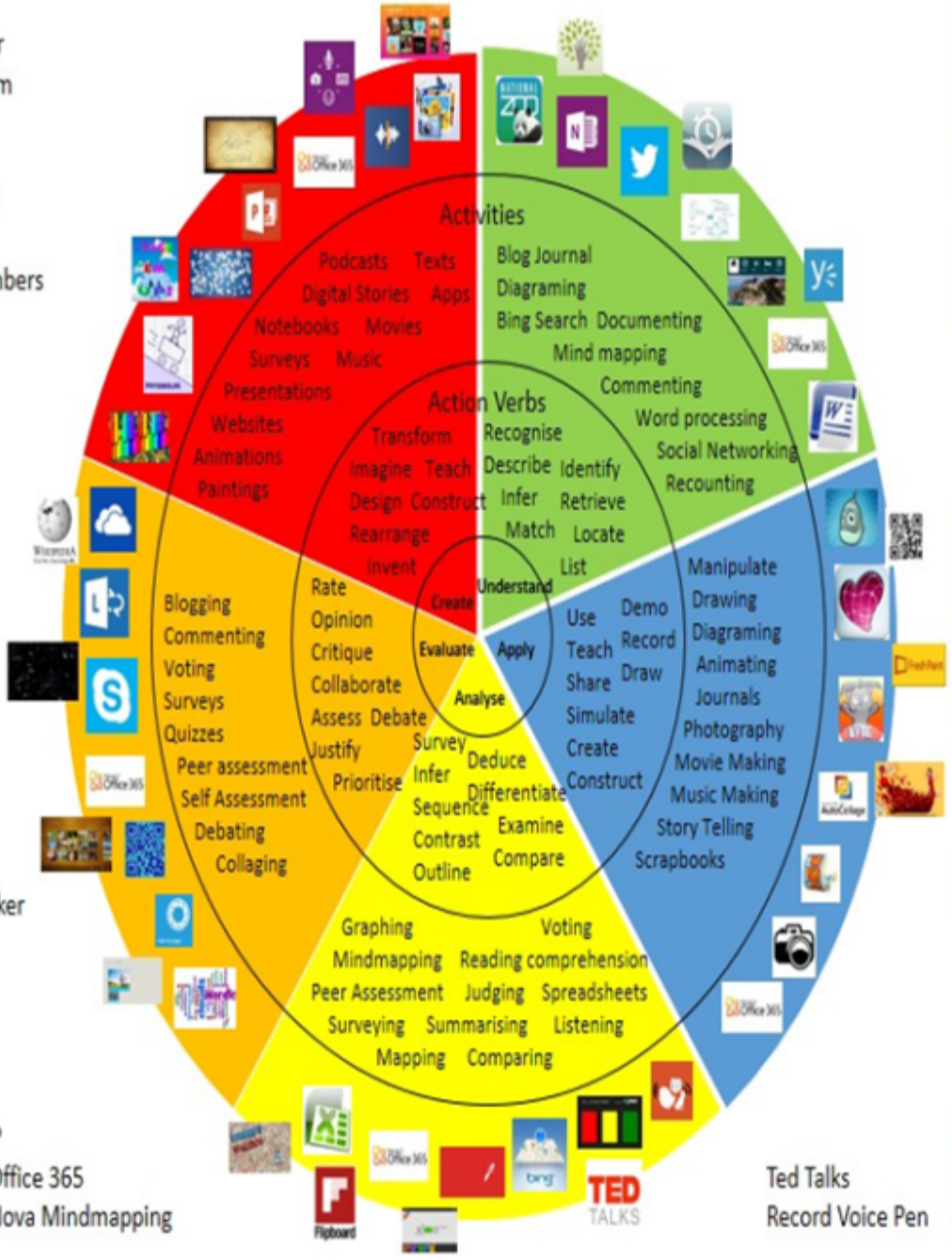
Please reference video links and websites listed under Ancillary Resources and Suggested Activities & Best Practices.

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

Win 8.1 Apps/Tools Pedagogy Wheel

Podcasts
Photostory 3
Kid Story Builder
Music Maker Jam
Paint A Story
Office 365
MS PowerPoint
Stack 'Em Up
NqSquared Numbers
Physamajig
Xylophone 8

Wikipedia
Skydrive
Lync
SkyMap
Skype
Office 365
Puzzle Touch
Easy QR
Memorylage
Life Moments
Word Cloud Maker



Where's Waldo?
MS Excel Office 365
Flipboard Nova Mindmapping

Ted Talks
Record Voice Pen

Originally taken from http://www.coetail.com/vzimmer/files/2013/02/iPadagogy-Wheel.001.jpg
And adapted for Windows 8.1 devices by Charlotte Beckhurst @CharBeckhurst

Alignment to 21st Century Skills & Technology

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.

CRP.K-12.CRP1	Act as a responsible and contributing citizen and employee.
CRP.K-12.CRP2	Apply appropriate academic and technical skills.
CRP.K-12.CRP4	Communicate clearly and effectively and with reason.
CRP.K-12.CRP6	Demonstrate creativity and innovation.
CRP.K-12.CRP7	Employ valid and reliable research strategies.
CRP.K-12.CRP8	Utilize critical thinking to make sense of problems and persevere in solving them.
CRP.K-12.CRP11	Use technology to enhance productivity.
CAEP.9.2.12.C.2	Modify Personalized Student Learning Plans to support declared career goals.
CAEP.9.2.12.C.3	Identify transferable career skills and design alternate career plans.
TECH.8.1.12.A	Technology Operations and Concepts: Students demonstrate a sound understanding of technology concepts, systems and operations.
TECH.8.1.12.A.3	Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
TECH.8.1.12.A.CS1	Understand and use technology systems.
TECH.8.1.12.A.CS2	Select and use applications effectively and productively.
TECH.8.1.12.B	Creativity and Innovation: Students demonstrate creative thinking, construct knowledge and develop innovative products and process using technology.
TECH.8.1.12.C	Communication and Collaboration: Students use digital media and environments to communicate and work collaboratively, including at a distance, to support individual learning and contribute to the learning of others.

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

21st Century/Interdisciplinary Themes that will be incorporated into this unit.

- 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

CRP.K-12.CRP1	Act as a responsible and contributing citizen and employee.
CRP.K-12.CRP2	Apply appropriate academic and technical skills.
CRP.K-12.CRP4	Communicate clearly and effectively and with reason.
CRP.K-12.CRP6	Demonstrate creativity and innovation.

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

12.9.3.IT.3	Demonstrate the use of cross-functional teams in achieving IT project goals.
12.9.3.IT.4	Demonstrate positive cyber citizenry by applying industry accepted ethical practices and behaviors.
12.9.3.IT.5	Explain the implications of IT on business development.
12.9.3.IT.6	Describe trends in emerging and evolving computer technologies and their influence on IT practices.

Differentiation

• Technology Resources • Teacher Tutoring • Peer Tutoring • Cooperative Learning Groups • Differentiated Instruction • Follow all

IEP Modifications/504 Plan

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.

I've also used task cards in another way. I've printed twelve task cards and told the students they could move on once they have mastered one card. I've made it clear that some students would get further then other in the course of one period and that I was ok with that. Stressing the non-competition part is important when trying to use this differentiation technique because otherwise they just try to get through as many cards as possible without actually understanding what they are doing.

Differentiation Include:

- 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)

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

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.
- **Develop** simple study guides to complement required materials.

Providing extra material such as videos or an interactive way to learn the material is best, You Tube or a smaller scaffolding exercise where they understand one or two parts of the whole concept.

- 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
- multiple test sessions
- multi-sensory presentation
- 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)

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.

eDictionaries

When used correctly, technology never fails to transform a classroom. Allow your students to use eDictionaries in your classroom to look up unknown words. Any regular dictionary will do, but eDictionaries allow students to hear the pronunciations of words. As you might imagine, this is extremely helpful for anyone learning a second language. If students encounter an unknown word, they can type it into an eDictionary, hear the word pronounced and either read or listen to the definition. Also, if they aren't sure how to to spell a word, many eDictionaries will allow them to speak the word to have the correct spelling displayed.

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

- teaching key aspects of a topic. Eliminate nonessential information
- using videos, illustrations, pictures, and drawings to explain or clarif
- 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 workpresented 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

Directions or Instructions

Make sure directions and/or instructions are given in limited numbers. Give directions/instructions verbally and in simple written format. Ask students to repeat the instructions or directions to ensure understanding occurs. Check back with the student to ensure he/she hasn't forgotten. It is a rare event for students at risk to be able to remember more than 3 things at once. Chunk your information, when 2 things are done, move to the next two.

A Strategy Summary

Planned interventions are far superior to remediation approaches. Always plan to address students at risk in your learning tasks, instructions, and directions. Try to anticipate where the needs will be and then address them. Intervene as much as possible to support students at risk. If your intervention strategies are working, continue to use them. If they're not working, plan for new interventions that will help students succeed.

Intervention Strategies that will be employed in the unit, using the ones identified below.

- 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 workpresented 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)

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

Unit Name: Binary Numbers

In this lesson, students will gain more familiarity with binary numbers. The lesson will transition away from the number systems that students created in the the circle-triangle-square activity, and begin to focus on representing numeric values using the binary number system. Though students have communicated with binary before, developing a number system is a little different. Previously, students mapped patterns of binary values to a small set of fixed messages. A number system is infinite, and also has rules for counting - or how to get from one value to the next.

NJSLS:

CSTA K-12 Computer Science Standards (2011)

CT - Computational Thinking

Computer Science Principles

2.1 - A variety of abstractions built upon binary sequences can be used to represent all digital data.2.3 - Models and simulations use abstraction to generate new understanding and knowledge.

Interdisciplinary Connection: Writing-analysis/evaluation; Reading-comprehension

Statement of Objective:

Number systems help us express and reason about quantities. Early number systems were merely a system of tallies that allowed humans to record and perform simple arithmetic with values. The number system we use today uses the concept of place value to allow us to express any value we wish by combining only 10 symbols (0, 1, 2 …). We therefore call it a “base 10” number system. When developing a number system for a computer, we only have two symbols available to us, corresponding with the two states of a single bit. However, the power of place value allows our binary or “base 2” number system to express any value we wish.

When using this binary representation of numbers, certain values (1, 2, 4, 8, 16, etc.) are seen repeatedly. When written in binary, these values are 1, 10, 100, 1000, 10000, and so on, and so are the incremental place

values in this binary number system.

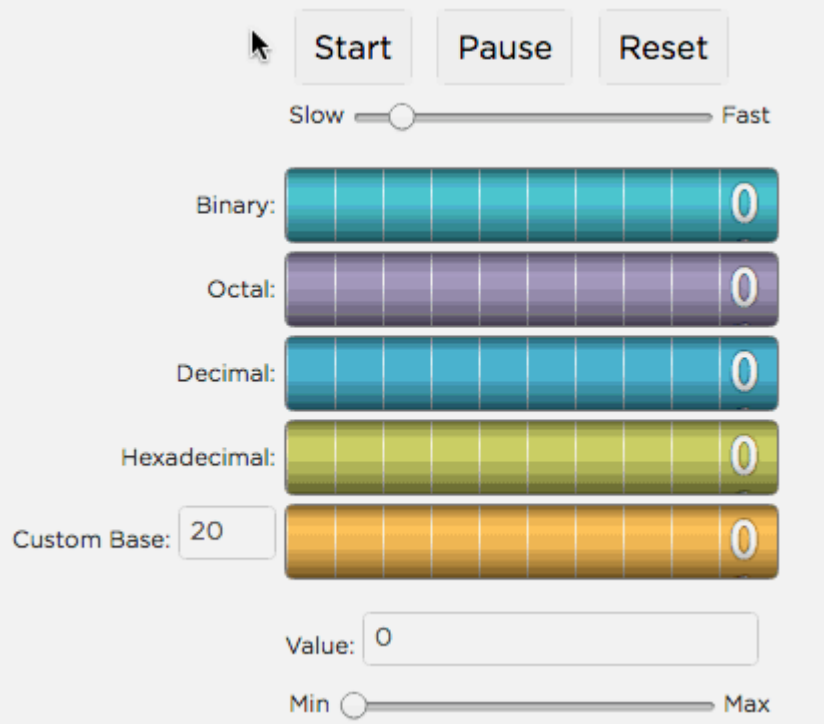
Students will be able to:

- Describe how to use bits to create a functioning number system
- Understand the relationship between the powers of 2 and the number of bits needed to express a number of a certain magnitude. e.g. How many bits do I need to represent the number “15”, or “32”, or “1492”?
- Determine, for a given number of bits, both the number of possible numbers that can be represented and also the range of those numbers

Anticipatory Set/Do Now:

Show the Binary Odometer (5 mins)

Teaching Tip



The binary odometer is simply meant to be an interactive tool for looking at numbers in various bases. You do not need to linger on it when introducing it. Students can make use of it when completing the activity guide.

One thing to note about the widget is that you can rearrange the various odometers to place ones you want next to each other.

Just click and drag to rearrange.

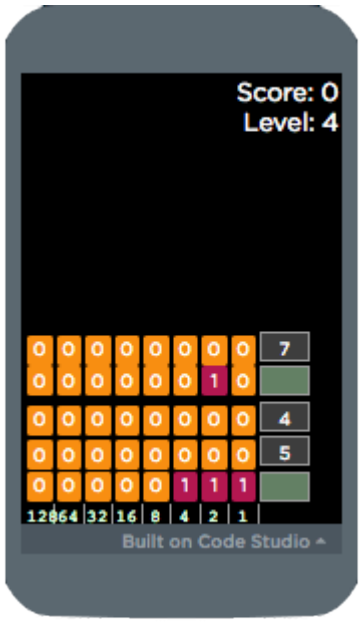
We'll keep filling in our Flippy Do's but you should also know about a helpful widget -- It was used in the opening demonstration -- called the "Binary Odometer"

- Go to the Binary Odometer - Code Studio (linked in Code Studio)
- This tool allows you to watch how counting is related across multiple number systems, binary, decimal and other number bases as well.
- Students should explore in pairs.
- Give students a few minutes to get acquainted with the Binary Odometer.

Tasks/Prompts to consider:

- "What's the largest number you can make in binary with the binary odometer?"
    - BIN: 1111111111 --> DEC: 1023
- "What happens when the odometer run out of numbers?"
    - Overflow! The binary odometer rolls back over to all zeros but the other numbers keep going up.

Complete the Binary Practice Activity Guide (15 mins)



Students should:

- Complete the Binary Practice - Activity Guide individually (They can work with a partner but each should complete their own sheet)
- Use their Flippy Do and Odometer as resources

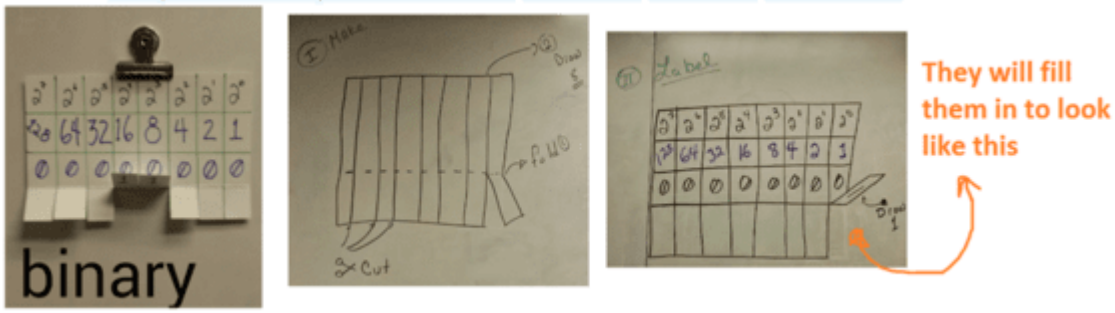
If students finish early, you may direct them to the Binary Game AppLab App - App.

Learning Activity:

Construct a Flippy Do (10 mins)

Remarks

For today's activity you will be creating your own "Flippy Do."



This is a tool that will allow you to quickly and easily translate between decimal numbers and binary numbers, or in other words, between the numbers we use every day and bits.

Teaching Tips

Demonstrating the Flippy Do: It may be necessary to demonstrate how values can be calculated by flipping up a "1" for each value required to arrive at the sum of values equal to the decimal number.

If students are having a difficult time understanding the rules of the system, remind them of the concept of place value and relate to base 10.

Make the Flippy Do

Student Assessment/CFU's: Observation,

Materials: Textbook, Worksheet, Computer

21st Century Themes and Skills:

Differentiation: Direct Instrution, Guided Instruction, extra time allowed for students with IEP

Task Challenges - Some students may have difficulty:
• reading and comprehending open/exploratory directions for constructing the Flippy Do
• comprehending information related to the powers of 2
• comprehending and using mathematical and binary principles
Adjustment Recommendations:
➢ Construct the Flippy Do as a class with significant guidance (i, 11, 20, 24, 29). AP
➢ Once the Flippy Do is constructed, have students go to the Binary Odometer, and have them experiment constructing numbers to see how the Odometer corresponds to the Flippy Do. Some students may need suggestions on numbers to try and build, and some may need additional explanation on how the Odometer and Flippy Do correspond to each other. (The Binary Odometer has several different numbering systems. This may cause confusion for some students.)

https://runestone.academy/runestone/books/published/csawesome/Unit1-Getting-Started/topic-1-2-java-intro.html