Unit 7 Recursion Copied from: AP Computer Science A, Copied on: 02/21/22

Content Area: **Technology**
Course(s): **Sample Course**
Time Period: **AprMay**
Length: **5-7 class periods**
Status: **Published**

Recursion - Computer Science A

Department of Curriculum and Instruction

Belleville Public Schools

Curriculum Guide

AP Computer Science A

Recursion

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

In this unit students will learn how to apply recursion in Java. We will cover several methods for recursion, including factorials, Fibonacci series, and the Tower of Hanoi game. Working code examples are provided.

Recursion: Calling Yourself

Imagine leaving a building that has a revolving door. Instead of exiting the building, you keep going around and around again, until someone tells you to exit (or you pass out from dizziness). In programming, a **recursive** method is like this: it calls itself over and over until something triggers its exit.

Recursion is considered a concise and advanced way of programming. These characterizations stem from the limited lines of code required as opposed to other, more verbose approaches.

Sometimes a problem can be solved by solving smaller or simpler versions of the same problem rather than attempting an iterative solution.

This is called recursion, and it is a powerful math and computer science idea.

In this unit, students will revisit how control is passed when methods are called, which is necessary knowledge when working with recursion.

Enduring Understanding

What real-world processes do you follow that are recursive in nature?

Why do programmers sometimes prefer using recursive solutions when sorting data in a large data set?

Recursion in computer science is a method where the solution to a problem depends on solutions to smaller instances of the same problem (as opposed to iteration).

The approach can be applied to many types of problems, and recursion is one of the central ideas of computer science.

Essential Questions

Which of the following conditions must be true in order to search for a value using binary search?

```
I. The values in the array must be integers.
II. The values in the array must be in sorted order.
III. The array must not contain duplicate values.
```

What is Recursion?

Which line in the method factorial contains the recursive call (the call to the same method)?

Exit Skills

Determine an appropriate program design to solve a problem or accomplish a task (not assessed). 3.B Write program code to define a new type by creating a class. 9.2 Writing Constructors for Subclasses 3.B Write

program code to define a new type by creating a class. 5.A Describe the behavior of a given segment of program code. 9.3 Overriding Methods 3.B Write program code to define a new type by creating a class. 5.D Describe the initial conditions that must be met for a program segment to work as intended or described. 9.4 super Keyword 1.C Determine code that would be used to interact with completed program code. 3.B Write program code to define a new type by creating a class. 9.5 Creating References Using Inheritance Hierarchies 3.A Write program code to create objects of a class and call methods. 5.B Explain why a code segment will not compile or work as intended. 9.6 Polymorphism 3.A Write program code to create objects of a class and call methods. 5.B Explain why a code segment will not compile or work as intended. 9.7 Object Superclass 1.C Determine code that would be used to interact with completed program code. 3.B Write program code to define a new type by creating a class

New Jersey Student Learning Standards (NJSLS-S)

CSTA Standards
Computing Systems • 3A-CS-03 Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors. • 3B-CS-02 Illustrate ways computing systems implement logic, input, and output through hardware components. ***Networks and the Internet*** • 3B-NI-03 Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology). ***Data and Analysis*** • 3A-DA-09 Translate between different bit representations of real-world phenomena, such as characters, numbers, and images. • 3A-DA-12 Create computational models that represent the relationships among different elements of data collected from a phenomenon or process. • 3B-DA-06 Select data collection tools and techniques to generate data sets that support a claim or communicate information. • 3B-DA-07 Evaluate the ability of models and simulations to test and support the refinement of hypotheses. ***Algorithms and Programming*** • 3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.

- 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-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.
- 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-14 Construct solutions to problems using student-created components, such as procedures, modules and/or objects.
- 3B-AP-20 Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (code documentation) in a group software project.
- 3B-AP-21 Develop and use a series of test cases to verify that a program performs according to its design specifications.
- 3B-AP-22 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).
- 3B-AP-23 Evaluate key qualities of a program through a process such as a code review.

Impacts of Computing

- 3A-IC-25 Test and refine computational artifacts to reduce bias and equity deficits.
- 3A-IC-26 Demonstrate ways a given algorithm applies to problems across disciplines.
- 3A-IC-29 Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.

8.2.12.B.1 Research and analyze the impact of the design constraints (specifications and limits) for a product or technology driven by a cultural, social, economic or political need and publish for review. 8.2.12.B.3 Analyze ethical and unethical practices around intellectual property rights as influenced by human wants and/or needs. 8.2.12.C.1 Explain how open source technologies follow the design process. 8.2.12.E.1 Demonstrate an understanding of the problem-solving capacity of computers in our world. 8.2.12.E.2 Analyze the relationships between internal and external computer components. 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). 8.2.12.E.4 Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and 8.2.12.B.1 Research and analyze the impact of the design constraints (specifications and limits) for a product or technology driven by a cultural, social, economic or political need and publish for review. 8.2.12.B.3 Analyze ethical and unethical practices around intellectual property rights as influenced by human wants and/or needs. 8.2.12.C.1 Explain how open source technologies follow the design process. 8.2.12.E.1 Demonstrate an understanding of the problemsolving capacity of computers in our world. 8.2.12.E.2 Analyze the relationships between internal and external computer components. 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). 8.2.12.E.4 Use appropriate terms in conditional statements). Support Standards

(CSTA): CPP.L2.04 Demonstrate an understanding of algorithms and their practical application CPP.L3A.05 Use Application Programming Interfaces (APIs) and libraries to facilitate programming solutions. CPP.L3A.08 Explain the program execution process. CRP Standards: 9.2.12.C.3 Identify transferable career skills and design alternate career plans. 9.2.12.C.5 Research career opportunities in the United States and abroad that require knowledge of world languages and diverse cultures. 9.2.12.C.6 Investigate entrepreneurship opportunities as options for career planning and identify the knowledge, skills, abilities, and resources required for owning and managing a business. 9.2.12.C.7 Examine the professional, legal, and ethical responsibilities for both employers and employees in the global workplace. CRP2 Apply appropriate academic and technical skills. CRP10 Plan education and career paths aligned to personal goals. CRP11 Use technology to enhance conditional statements).

CPP.L2.04 Demonstrate an understanding of algorithms and their practical application CPP.L3A.05 Use Application Programming Interfaces (APIs) and libraries to facilitate programming solutions. CPP.L3A.08 Explain the program execution process. CRP Standards: 9.2.12.C.3 Identify transferable career skills and design alternate career plans. 9.2.12.C.5 Research career opportunities in the United States and abroad that require knowledge of world languages and diverse cultures. 9.2.12.C.6 Investigate entrepreneurship opportunities as options for career planning and identify the knowledge, skills, abilities, and resources required for owning and managing a business. 9.2.12.C.7 Examine the professional, legal, and ethical responsibilities for both employers and employees in the global workplace. CRP2 Apply appropriate academic and technical skills. CRP10 Plan education and career paths aligned to personal goals. CRP11 Use technology to enhance.

TECH.8.2.12.E	Computational Thinking: Programming: Computational thinking builds and enhances problem solving, allowing students to move beyond using knowledge to creating knowledge.
TECH.8.2.12.E.1	Demonstrate an understanding of the problem-solving capacity of computers in our world.
TECH.8.2.12.E.2	Analyze the relationships between internal and external computer components.
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.WHST.11-12.1.A	Introduce precise, knowledgeable claim(s), establish the significance of the claim(s), distinguish the claim(s) from alternate or opposing claims, and create an organization that logically sequences the claim(s), counterclaims, reasons, and evidence.
9-12.HS-ETS1-1.1.1	Analyze complex real-world problems by specifying criteria and constraints for successful solutions.
9-12.HS-ETS1-4.4.1	Models (e.g., physical, mathematical, computer models) can be used to simulate systems and interactions— including energy, matter, and information flows— within and between systems at different scales.
	Modeling

Learning Objectives

Apply recursive search algorithms to information in String, 1D array, or ArrayList objects.

Data must be in sorted order to use the binary search algorithm.

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

Suggested Activities & Best Practices

Personal Progress Check 10 Multiple-choice: ~10 questions Free-response: 1 question § Methods and Control Structures (recursive and non-recursive solutions allowed)

10.1 Sharing and responding Provide students with the pseudocode to multiple recursive algorithms, and have students write the base case of the recursive methods and share it with their partner. The partner should then provide feedback, including any corrections or additions that may be needed. 2 10.1 Look for a pattern Provide students with a recursive method and several different inputs. Have students run the recursive method, record the various outputs, and look for a pattern between the input and related output. Ask students to write one or two sentences as a broad description of what the recursive method is doing. 3 10.2 Code tracing When looking at a recursive method to determine how many times it executes, have students create a call tree or a stack trace to show the method being called and the values of any parameters of each call. Students can then count up the number of times a statement executes or a method is called.

Assessment Evidence - Checking for Understanding (CFU)

Runestone Academy: AP CSA—Java Review: 12—Recursion - Practice-It!: BJP4 Chapter 12: Recursion—Self Check 12.3–12.6, 12.13–12.15 § Coding Bat Java: Recursion

Formative Evaluations:	**Summative Evaluations:**
Formative Assessment with polling codeIt! Nows Quizzes Long Programs (LP)/Lab Work Components of AP approved Lab	Unit Test/ReTest-Summative Assessment

Web-based Assessment-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://runestone.academy/runestone/books/published/csawesome/Unit10-Recursion/toctree.html?highlight=recursion

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.

https://chortle.ccsu.edu/CS151/cs151java.html

Technology Infusion

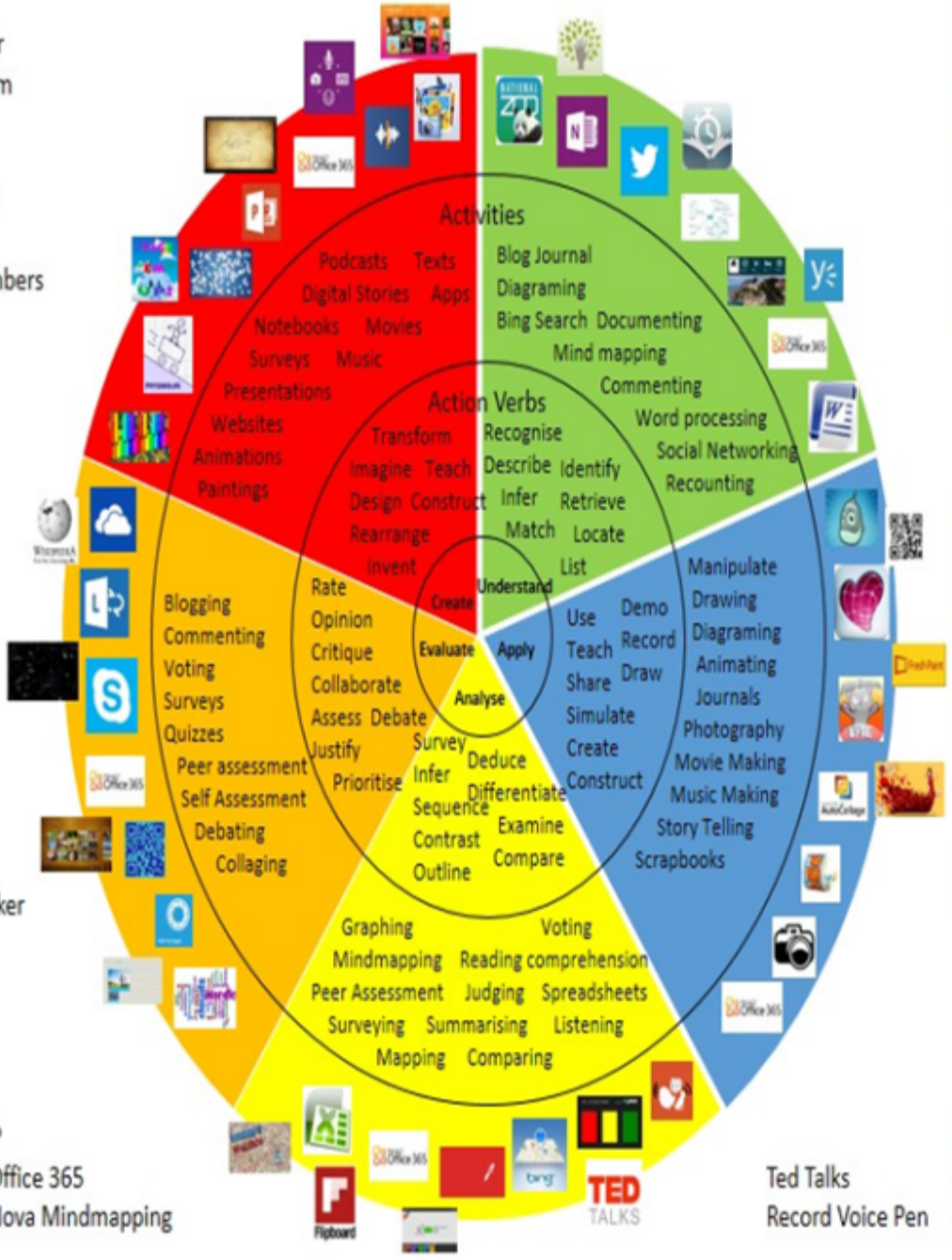
Technology Infusion and/or strategies include chromebooks online materials google/powerpoint slides

Technology Infusion and/or strategies are integrated into this unit to enhance learning

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

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.

CAEP.9.2.12.C	Career Preparation
CAEP.9.2.12.C.1	Review career goals and determine steps necessary for attainment.
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.

21st Century Skills/Interdisciplinary Themes

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

Functions

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.

Connections to Expressions, Equations, Modeling, and Coordinates.

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

CAEP.9.2.12.C	Career Preparation
CAEP.9.2.12.C.1	Review career goals and determine steps necessary for attainment.
CAEP.9.2.12.C.2	Modify Personalized Student Learning Plans to support declared career goals.

Differentiation

Exemplar: Study guides

Differentiation in a lesson lies within content, process, and/or product.

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 could be employed in this unit, using the ones identified below.

- 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)

Exemplar:

Students are assigned a partner for tutoring and assistance in class

Students are given fewer choices on quizzes and tests.

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

Assignments

Exemplar: The student at risk will need many assignments modified or reduced. Always ask yourself, "How can I modify this assignment to ensure the students at risk are able to complete it?" Sometimes you'll simplify the task, reduce the length of the assignment or allow for a different mode of delivery. For instance, many students may hand something in, the at-risk student may make jot notes and give you the information verbally, or it just may be that you will need to assign an alternate assignment.

- 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)

Exemplar:

Students will create a blog or social media page a topic of their choice within 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.

- 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

Using the template below, please develop a **Sample Lesson** for the first unit only.

Unit Name:

NJSLS:

Interdisciplinary Connection:

Statement of Objective:

Anticipatory Set/Do Now:

Learning Activity:

Student Assessment/CFU's:

Materials:

21st Century Themes and Skills:

Differentiation/Modifications:

Integration of Technology: