Wayne
Township Public Schools

Mathematics
Grade 11-12
AP Computer Science A

Dr. Mark Toback, Superintendent
Committee: Thomas Grasso

This curriculum may be modified through varying techniques, strategies, and materials as per an individual student's Individualized Educational Plan (IEP)

Wayne School District
Curriculum Format

Content Area/ Grade Level/ Course:	**Mathematics 10-12 AP Computer Science**
Unit Plan Title:	**Unit 1. Hardware**
Time Frame	**2 weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
Computer Systems	CS
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 1. Hardware

A. What is the difference between hardware and software?
B. RAM/ ROM
C. How basic computing works
D. How computers store information
E. Finding a computer on a network
F. Very basic intro to Java

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible
8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
8.1.12.CS.3: Compare the functions of application software, system software, and hardware.
8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.
9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
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).
9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)
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.
9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
RST.11-12.3. Follow precisely a complex multistep procedure when carrying out experiments, taking measurements, or performing technical tasks; analyze the specific results based on explanations in the text.
RST.11-12.4. Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 11-12 texts and topics.
RST.11-12.9. Synthesize information from a range of course (e.g., texts, experiments, simulations) into a coherent understanding of a process, phenomenon, or concept, resolving conflicting information when possible.
WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.

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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.

CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.

CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.

HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.

HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.

HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.

HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is the difference between hardware and software?
- What is ROM and RAM?
- How do computers store and access data?
- What is an input and output device?
- What is an IP address?

Enduring Understandings

- Hardware is a tangible device. Software is the instructions which make the device do a task.
- ROM is read- only memory. This memory is usually hardwired into a computer. RAM is Random access memory. This memory is volatile and is constantly changing.
- Computers store data using a series of states called binary. This can be represented theoretically by ones and zeros or physically using anything that can be placed in a row and put into two distinct states. Magnetism, electricity, and pits are common ways computers use this idea.
- An input device allows a user to enter information. A keyboard and mouse are common ones. An output device allows a user to get information from the computer. A screen and a printer are common ones.
- An IP address is the computer's virtual location on a network.

In this unit plan, the following 21st Century themes and skills are addressed.

Check all that apply. **21st Century Themes**		*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.* **21st Century Skills**	
X	**Global Awareness**	**E,T,A**	**Creativity and Innovation**
	Environmental Literacy	**E,T,A**	**Critical Thinking a Problem Solving**
	Health Literacy	**E,T,A**	**Communication**
	Civic Literacy	**E,T,A**	**Collaboration**
X	**Financial, Economic, Business, and Entrepreneurial Literacy**		

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will know the difference between hardware and software.
- Students will know what ROM and RAM are and how the computer uses a combination of the two to run.
- Students will understand how computer store data and can retrieve it very quickly when needed.
- Students will know the difference between an input device and an output device.
- Students will understand how an IP address works and how to locate themselves and other computers on a network.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****

Students will work from a program specification to build
their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer b will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Wayne School District
Curriculum Format

Content Area/ Grade Level/ Course:	**Mathematics** **10-12** **Java**
Unit Plan Title:	**Unit 2: Objects and Primitive Data**
Time Frame	**3 Weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED

9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking

ELA-Literacy:WHST Writing: Text Types and Purposes

ELA-Literacy: RST Reading: Craft and Structure

HS-ETS1 Engineering Design

Unit Summary

Unit 2: Objects and Primitive Data

- Introduce data and data types.
- Introduce the idea of objects, classes, and methods.
- Examine different types of data and what they are used for.
- Introduce classes students must know specific to the AP exam.
- Create objects.

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible
8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.
9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
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).
9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)
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.
9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.
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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.
CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.
HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.
HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

Unit 2: Objects and Primitive Data

- What is a primitive?
- What is an object?
- How do we declare and use each in Java?
- What are some common classes that appear on the AP exam?

Enduring Understandings

- A primitive is a simple piece of data. It has an established size and use.
- An object is more dynamic than a primitive. It can contain many different primitives and has an unknown size.
- Primitive example: int, boolean, char
- Objects use the "new" operator to create a reference to space in memory where an object exists.
- Commonly used AP classes are String, Scanner, DecimalFormat, and Random.

In this unit plan, the following 21st Century themes and skills are addressed.

Check all that apply. **21st Century Themes**		*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.* **21st Century Skills**	
X	**Global Awareness**	E,T,A	**Creativity and Innovation**
	Environmental Literacy	E,T,A	**Critical Thinking and Problem Solving**
	Health Literacy	E,T,A	**Communication**

	Civic Literacy	**E,T,A**	**Collaboration**
X	**Financial, Economic, Business, and Entrepreneurial Literacy**		

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will develop methods for creating possible solutions, modeling and testing solutions, and modifying proposed design in the solution of a technological problem using hands-on activities.
- Students will learn to create and use primitives and objects.
- Students will show they can use these types.
- Students will understand there are many differences between objects and primitives, more of which will show up later.
- Students will practice and understand the Java syntax for using these types.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****

Students will work from a program specification to build their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion o program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer based will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities

Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
B Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	
• **http://www.state.nj.us/education/cccs/** • **http://www.corestandards.org/ELA-Literacy** • **http://www.state.nj.us/education/cccs/**	

Wayne School District

Curriculum Format

Content Area/ Grade Level/ Course:	**Mathematics / Computer Science 10-12 Java**
Unit Plan Title:	**Unit 3. Loops and Conditionals**
Time Frame	**3 Weeks**
Anchor Standards/Domain*	***i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10**

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 3. Loops and Conditionals

A. Introduction to boolean logic
B. Using logic statements to run if, for, and while conditionals.
C. Understanding the syntax and program counter.
D. Steps to program creation
E. Understanding flowcharts.

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.

8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.

8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.

8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.

8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects

8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.

8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible

8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.

8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.

9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).

9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).

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).

9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)

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.

9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).

WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.

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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.

CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.
HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.
HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is boolean logic and how do computers use it?
- What is the difference between an if statement and a loop?
- How can ifs and loops be used to create complex code?
- What is the syntax of an if, for, and while? How does this affect the program counter?
- What are the four steps to the program creation process?
- How can flowcharts be used to show logic?

Enduring Understandings

- Boolean logic is logic where variables have two values, or states, true or false. This idea is constantly used in computing and is the backbone of binary.
- An if statement allows the computer to "choose" to run code based on a boolean statement. True, it runs, false, it does not. A loop allows statement repetition without rewriting code. True, a statement repeats, false, it does not.
- If statements and loops are used in all code. Anytime the programmer needs a "choice" or something to repeat, these statements are needed. Large projects cold contain hundreds of these statements.
- If (condition) {} is the syntax for an if. while(condition) {} is a while loop. for(initializer; condition; increment){} is a for loop.
- The four steps in program creation are understand the problem, analyze the problem, code, and test. The last two steps are often repeated.

- A flowchart can graphically display how a problem is solved. It can be easily read by others because it takes the idea of logic and makes it concrete.

In this unit plan, the following 21st Century themes and skills are addressed.

Check all that apply.
21st Century Themes

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

Indicate whether these skills are ***E****-Encouraged,* ***T****-Taught, or* ***A****-Assessed in this unit by marking* ***E, T, A*** *on the line before the appropriate skill.*
21st Century Skills

E,T,A	**Creativity and Innovation**
E,T,A	**Critical Thinking and Problem Solving**
E,T,A	**Communication**
E,T,A	**Collaboration**

Student Learning Targets/Objectives (Students will know/Students will understand)

1. Students will analyze a given technological product, system, or environment to understand how the engineering design process and design specification limitations influenced the final solution.
2. Students will create and test their own programs.
3. Students will understand boolean statements.
4. Students will understand logic and how it is used it to help solve problems.
5. Students will understand the difference between an if and a loop and when each is needed.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****

Students will work from a program specification to build their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer based will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Wayne School District

Curriculum Format

Content Area/ Grade Level/ Course:	**Mathematics 10 -12 Java**
Unit Plan Title:	**Unit 4. Classes, Objects, and Methods**
Time Frame	**4 Weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED

9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking

ELA-Literacy:WHST Writing: Text Types and Purposes

ELA-Literacy: RST Reading: Craft and Structure

HS-ETS1 Engineering Design

Unit Summary

Unit 4. Classes, Objects, and Methods

- A closer look at classes, objects, and methods.
- Organized code
- Creating objects from classes and why it is important
- Using the constructor
- Code reuse
- Keeping methods with their proper classes
- Using objects is problem solving

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
8.1.12.DA.3: Translate between decimal numbers and binary numbers.
8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device.
8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible
8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.
9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
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).
9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)
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.
9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.
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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented
CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.
CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.
HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.
HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is the difference between a class and an object? What is a method?
- What are the accepted formats for Java code?
- Why should code be split into and stored in classes?
- How do we create an object?
- What is a constructor and why is it important?
- How do we know which methods goes with which class?
- How can we use objects to help solve a problem?

Enduring Understandings

Unit 4. Classes, Objects, and Methods

- A class is a blueprint that objects can be created from using the "new" keyword. Methods are actions that these objects can perform.
- Curly braces can either appear on the same line or their own line. Code within the curly brace should be tabbed appropriately.
- Classes allow code reuse which is a main component of object oriented programming. (OOP)
- The "new" keyword allows us to create an object.
- The constructor is the object's setup statement. It sets the variables to the appropriate values give to the object.
- Methods should always pertain to the class they are in. If a method does not, it might belongs in another class.

- Objects can be instantiated and used to represent one instance of a class. The constructor can take general code and apply it to a specific problem.

In this unit plan, the following 21st Century themes and skills are addressed.

	Check all that apply. **21st Century Themes**		*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.* **21st Century Skills**
X	**Global Awareness**	**E,T,A**	**Creativity and Innovation**
	Environmental Literacy	**E,T,A**	**Critical Thinking and Problem Solving**
	Health Literacy	**E,T,A**	**Communication**
X	**Civic Literacy**	**E,T,A**	**Collaboration**
X	**Financial, Economic, Business, and Entrepreneurial Literacy**		

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will be able to create a class, object, and methods.
- Students will understand why class and objects are important to OOP
- Students will use the constructor to set up their objects.
- Students will know the proper coding format.
- Students will be able to identify methods given a class name and a class name given its methods.
- Students will understand when they need to instantiate a new object.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****
Students will work from a program specification to build their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date. Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer based will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment (Learning Style Assessment results) • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons

Resources

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Wayne School District
Curriculum Format

Content Area/ **Grade Level/** **Course:**	**Mathematics** **10-12** **AP Computer Science**
Unit Plan Title:	**Unit 5. Enhancing Classes**
Time Frame	**5 weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 5. Enhancing Classes

- Aliases
- Interfaces
- Commonly used Java interfaces
- Passing objects as parameters

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
8.1.12.DA.3: Translate between decimal numbers and binary numbers.
8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device.
8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible
8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.
9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
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).
9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)
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.
9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.
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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.

CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.

CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.

HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.

HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.

HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.

HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is an alias and how does it affect code?
- How does an alias and a primitive behave differently when passed as a parameter?
- How can you compare two objects? What does == really mean?
- Why do we use interfaces and how do you make one?
- What are common interfaces?

Enduring Understandings

- An alias is two variables that reference the same object. If one changes the object it cold be changed for both.
- A primitive creates a copy of itself when passed as a parameter. The copy and original no longer are connected. An object creates an alias when it is passed. There is still a connection between the original and the alias. This means a programmer could unintentionally change an object's states.
- The == operator in Java checks to see if two objects are aliases of each other, meaning they reference the same object. With primitives, it checks to see if they are the same value.
- An interface takes classes with common actions, or methods, and creates a connection between them. An interface has no actual methods bodies and cannot be instantiated. They are used more for polymorphism.
- Commonly used interfaces on the AP exam are Comparable and List.

In this unit plan, the following 21st Century themes and skills are addressed.	
Check all that apply. **21st Century Themes**	*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.* **21st Century Skills**

	X	Global Awareness		E,T,A	Creativity and Innovation
		Environmental Literacy		E,T,A	Critical Thinking a Problem Solving
		Health Literacy		E,T,A	Communication
		Civic Literacy		E,T,A	Collaboration
	X	Financial, Economic, Business, and Entrepreneurial Literacy			

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will know what an alias is.
- Students will understand the difference between passing a primitive and an object.
- Students will know when they need to use the == operator and when to use a method to compare objects.
- Students will know how make with own interface.
- Students will know the commonly used interfaces.

Assessments (Pre, Formative, Summative, Other) *Denote required common assessments with an **

Students will work from a program specification to build
their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer b will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Content Area/ Grade Level/ Course:	**Mathematics 10-12 AP Computer Science**
Unit Plan Title:	**Unit 6. Arrays**
Time Frame	**6 weeks**

Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 6. Arrays

- An introduction to arrays
- Arrays of primitives and objects
- Sorting
- Searching
- Algorithms to sort and search
- 2D arrays
- ArrayLists and data structure attributes.

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.

8.1.12.DA.3: Translate between decimal numbers and binary numbers.

8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device.

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.

8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.

8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.

8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.

8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects

8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.

8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible

8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.

8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.

9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).

9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).

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).

9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)

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.

9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).

WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.

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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.

CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.

CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.

HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.

HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.

HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.

HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is an array and how does it help us keep data organized?
- Can you use arrays to organize objects?
- What are the two linear sorting algorithms on the AP exam and their Big Oh?
- What are the two searching algorithms on the AP exam and their Big Oh?
- Can you understand how they work?
- How can you use a 2D array to represent data such as a grid?
- What is the difference between an array and an ArrayList? When should you use them?

Enduring Understandings

- An array assigns index values to a group of objects or primitives so you can refer to them by their number rather than an exact variable name. This allows loops to be used to reference a group of data in one line.
- Arrays can be used to organize an data types.
- The two sorting algorithms are Selection and Insertion sort.
- The two searches are the linear search and binary search.
- Using a 2D array to organize data that is in a table format is much easier than using a single array. The 2D array allows access much like a cartesian plane with each cell having an x and a y value.
- An array is used if the programmer knows exactly how many elements they need in an array. This amount cannot change. An Arraylist is dynamic and can change size.

In this unit plan, the following 21st Century themes and skills are addressed.	
Check all that apply.	*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.*

21st Century Themes			21st Century Skills	
	X	Global Awareness	E,T,A	Creativity and Innovation
		Environmental Literacy	E,T,A	Critical Thinking a Problem Solving
		Health Literacy	E,T,A	Communication
		Civic Literacy	E,T,A	Collaboration
	X	Financial, Economic, Business, and Entrepreneurial Literacy		

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will understand when they need an array.
- Students will know how to access, organize, swap, change, and output array elements.
- Students will understand both sorting algorithms and how to create them from scratch.
- Students will understand both searching algorithms and how to create them from scratch.
- Students will know the Big Oh for all algorithms.
- Students will know when to use a 2D array to represent data.
- Students will know when to use an arrayList over an array.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****

Students will work from a program specification to build their own creative solutions to problems. By its very nature the course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer b will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Content Area/ Grade Level/ Course:	**Mathematics 10-12 AP Computer Science**
Unit Plan Title:	**Unit 7. Hierarchies**
Time Frame	**6 weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 7. Hierarchies

- An introduction to hierarchies. The keyword “extends”
- Organizing classes and their methods. Setting up hierarchies.
- Promoting methods and variables
- Review of casting
- Review of interfaces
- Polymorphism

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.

8.1.12.DA.3: Translate between decimal numbers and binary numbers.

8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device.

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.

8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.

8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.

8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.

8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects

8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.

8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible

8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.

8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.

9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).

9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).

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).

9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)

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.

9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).

WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.

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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.
CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.
CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.
HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.
HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- When should a class extends another?
- How should classes be organized into a useful OOP format?
- What methods and variables should be in a parent class?
- When should a programmer cast?
- When should a programmer create an interface so that "unrelated" code can be accessed together?
- What is polymorphism and why is it a very useful tool in OOP coding?

Enduring Understandings

- Classes that are related in some way might use the extends keyword. If there is a "is a" relationship between the classes this is used.
- OOP is set up for code reuse. The idea is that once someone creates code and moves on to the next project, the old project can still be useful. We see it constantly today in the form of new versions of products, but there are many other uses.
- Any method or variable that are in all siblings of a parent should be promoted to reside in the parent.
- A programmer should cast when narrowing conversion.
- When there are common methods between two unrelated classes, a programmer might want to create an interface so that they can use either one at any given time by casting.
- Polymorphism allows a variable to act like different objects at different times in a project. It allows the programmer to reuse old code and make connection between classes.

In this unit plan, the following 21st Century themes and skills are addressed.	
Check all that apply.	*Indicate whether these skills are* ***E****-Encouraged,* ***T****-Taught, or* ***A****-Assessed in this unit by marking* ***E, T, A*** *on the line before the appropriate skill.*

21st Century Themes			21st Century Skills		
	X	Global Awareness		E,T,A	Creativity and Innovation
		Environmental Literacy		E,T,A	Critical Thinking a Problem Solving
		Health Literacy		E,T,A	Communication
		Civic Literacy		E,T,A	Collaboration
	X	Financial, Economic, Business, and Entrepreneurial Literacy			

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will create their own hierarchies and make them function.
- Students will analyze code and make it efficient using hierarchies.
- Students will understand how classes should relate to each other in an organized fashion.
- Students will use polymorphism to create rich OOP code which can be reused later.

Assessments (Pre, Formative, Summative, Other) *Denote required common assessments with an **

Students will work from a program specification to build their own creative solutions to problems. By its very nature t course is highly differentiated and each student is free to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer b will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities

Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons

Resources

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/

Content Area/ Grade Level/ Course:	**Mathematics 10-12 AP Computer Science**
Unit Plan Title:	**Unit 8. Recursion**
Time Frame	**4 weeks**

Anchor Standards/Domain*   *i.e: ELA: reading, writing i.e.: Math: Number and Operations in Base 10

Data and Analysis	DA
Algorithms and Programming	AP
Engineering Design	ED
21st Century Life Skills	9.1.12.F
21st Career Awareness, Exploration & Preparation	9.3.12.C
Career & Technical Education	9.4.12.A
Career & Technical Education	9.4.12.B

Unit Summary

Unit 8. Recursion

- An introduction to recursion
- A look at iteration and recursion
- The base case
- Using recursion to solve problems.
- Merge Sort and Quick Sort

Standard Number(s)

i.e: Math: 3.NBT.1 i.e.: RL 8.1

Math: 3.NBT.1 i.e.: RL 8.1

8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and Accessible
8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
8.2.12.ED.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback.
9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
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).
9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a)
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.
9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
WHST.11-12.4. Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
WHST.11-12.5. Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.
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.

CCSS.ELA-Literacy.WHST.11-12.1e Provide a concluding statement or section that follows from or supports the argument presented.

CCSS.ELA-Literacy.RST.11-12.4 Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to *grades 11–12 texts and topics*.

CCSS.ELA-Literacy.RST.11-12.5 Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.
CCSS.ELA-Literacy.RST.11-12.6 Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, identifying important issues that remain unresolved.
HS-ETS1-1. Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
HS-ETS1-2. Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
HS-ETS1-3. Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.
HS-ETS1-4. Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Essential Question(s)

- What is recursion and when should it be used?
- Can all problems be solved using recursion?
- What is the base case and how does it stop an infinite recursion?
- What types of problems can be solved using recursion?
- What is the Merge Sort and Quick Sort?

Enduring Understandings

- Recursion is an elegant method of solving certain problems. It's a method that calls upon itself a finite number of times to solve a problem.
- Not all problems can be solved recursively, but all problems that can be solved recursively can ALSO be solved iteratively.
- The base case is an absolute end to the calling of itself. Without the base case, the recursion will call itself forever.
- Traversing mazes, Towers of Hanoi, fractals, and sorting are a few classic recursive problems.
- The merge and quick sort use recursion to efficiently sort data in many different set ups. Ironically, these divide and conquer algorithms' worst case is when the list is nearly in order (Quick sort).

In this unit plan, the following 21st Century themes and skills are addressed.					
Check all that apply. **21st Century Themes**			*Indicate whether these skills are **E**-Encouraged, **T**-Taught, or **A**-Assessed in this unit by marking **E, T, A** on the line before the appropriate skill.* **21st Century Skills**		
	X	**Global Awareness**		**E,T,A**	**Creativity and Innovation**

	Environmental Literacy	E,T,A	Critical Thinking a Problem Solving
	Health Literacy	E,T,A	Communication
	Civic Literacy	E,T,A	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		

Student Learning Targets/Objectives (Students will know/Students will understand)

- Students will be able to work with problems where recursion is a useful tool.
- Students will understand all recursive algorithms need a base case to stop the calls.
- Students will be able to analyze recursion problems and predict their output.
- Student will know the two new sorts and their Big Oh values.

Assessments (Pre, Formative, Summative, Other) ***Denote required common assessments with an ****

Students will work from a program specification to build their own creative solutions to problems. By its very nature th
course is highly differentiated and each student is free
to develop advanced features and enhancements beyond those required for a basic solution. The introduction of GUI (Graphic User Interfaces) early in the course further increases student options for creating exceptional graphics based applets and applications in addition or in place of a console based solution. Backwards Design in the discussion of a program solution will be featured regularly as we carefully consider which type of designs will provide for optimal solutions given the tools that students have acquired to date.

Formative and summative evaluations in the form of worksheets, quizzes and tests - both paper based and computer b
will be utilized as needed to reinforce learning objectives and evaluate student progress.

Teaching and Learning Activities	
Activities	• Lecture and class discussion. • Video and multimedia presentations. • Build a console, applet and GUI based application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
Differentiation Strategies	• Individual and collaborative research, design and problem solving • Student interest and skill level assessment • Individual, small group, and large group instruction • Media presentations and guest speakers • Student presentations and Flipped Lessons
Resources	

- http://www.state.nj.us/education/cccs/
- http://www.corestandards.org/ELA-Literacy
- http://www.state.nj.us/education/cccs/