

Wayne

Township Public Schools

Mathematics/Computer Science

Grades 9-12

Foundations of Computer Programming

Full Year

Dr. Mark Toback, Superintendent

Committee: Bruce McNutt

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

Approved by the Wayne Township Board of Education at the regular meeting held on **Month Day, 2021.**

Wayne School District Curriculum Format

Content Area/ Grade Level/ Course:	Mathematics/Technology 9-12 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 1: Programming Fundamentals – Creating Programs, Syntax, Data, Variables, Input, Output
Time Frame	15 days
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra	
<p>8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.</p> <p>8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.</p> <p>9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking</p>	
Unit Overview	

- Introduction to the concept of Computer Science and Computer Programming
- Introduction to Console Application Programming
- Input and Output in programming
- Variables and Data types
- Basic Mathematical Operations with numerical data types
- Basic String operations
- Algorithms
- Debugging and error checking
- Basic Problem Solving

Standard Number(s)

* i.e: **Math: F.LE.A.4**

i.e.: **NJSLSA.R4.**

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
 - 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.
 - NJLSA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
 - SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
 - 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- What is computer science and computer programming?
- What are the positive and negative impacts of technology on society?
- How can we classify data and what are data types?
- How do we use variables to store various data types?
- How do we display information to the user in a console application?
- How do we receive input from a user in a console application?
- How can we use mathematical operations in our code?
- What are string operations and how are they useful?
- What is an algorithm?
- How can we create a program to solve basic problems?

Enduring Understandings

- Technology and programming languages are constantly evolving.
- Technology has impacted virtually every industry and field of endeavor in our society.
- Programs are fundamentally very specific and literal instructions with specific syntax.
- Variables and specific Data types play an important role in input/output in programs.
- There are various operations associated with the different data types.
- A program is an algorithm that is used to solve a particular problem or perform a given task.

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

Check all that apply. 21 st 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. 21 st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication

	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		
Student Learning Targets/Objectives (Students will know/Students will understand)			
<ul style="list-style-type: none"> • Understand the basic principles behind computer science and the basic ideas of computer programs. • Understand variables and data types and apply them to simple programs • Apply the basic concepts of input/output in console/command line programs • Use the various operations associated with data types. • Understand the basic concepts of problem solving and apply them to creating algorithms • Identify several uses for computer programming in modern society. • Understand the idea of debugging and fixing a program 			
Assessments (Pre, Formative, Summative, Other)		<i>Denote required common assessments with an *</i>	
<p>This course utilizes Project Based Learning. Students will learn about various coding concepts and will have to apply their knowledge by performing various programming exercises. Projects will be assigned when the students have learned enough concepts to develop a program to complete a task.</p> <p>Formative and summative evaluations in the form of programming task, quizzes, and tests - both paper based and computer based will be utilized as needed to reinforce learning objectives and evaluate student progress.</p>			
Teaching and Learning Activities			

<p><i>Activities</i></p>	<ul style="list-style-type: none"> • Lecture and class discussion. • Video and multimedia presentations. • Build a console application. • Review and extend program functionality. • Group and collaborative work. • Student presentations of projects.
<p><i>Differentiation Strategies</i></p>	<ul style="list-style-type: none"> • 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 <p>Differentiation Strategies for Special Education Students Differentiation Strategies for Gifted and Talented Students Differentiation Strategies for ELL Students Differentiation Strategies for At Risk Students</p>
<p><i>Honors</i></p>	<p>N/A</p>
<p>Resources</p>	
<ul style="list-style-type: none"> • 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 9-12 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 2: Program Flow, Decisions, and Loops
Time Frame	30 days
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra	
<p>8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.</p> <p>8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.</p> <p>9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking</p>	
Unit Overview	
<ul style="list-style-type: none"> ● Program flow and changing the flow of code. ● Syntax of “if”, “if else”, “if else if”, “for”, and “while” loops. ● Boolean values, Conditional operators and compound operators. ● Use of conditionals and loops. ● Organizing code and blocking statements using correct syntax ● The need for disrupting the flow of control in programs. ● Problem solving using loops and conditionals ● Combining statements to achieve a goal. 	
Standard Number(s) * i.e: Math: F.LE.A.4 i.e.: NJSLSA.R4.	

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJSLA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- Why is it important to be able to change the flow of code?
- What is a control statement?
- What are the conditional operators and how are they incorporated in the different control structures?
- How do you use an if, if else, and if else if statements?
- How do you use a for loop and while loop?
- What are the components of a for loop?
- How can these control structures be used in solving problems/creating programs?

Enduring Understandings

- Flow of a program can be disrupted in many useful ways.
- Conditional operators play a key role in the various control structures.
- Compound operators can help reduce the lines of code and make programs more efficient.
- Programmers can use loops to eliminate lines of code and make programs more efficient.
- Programmers can use conditional statements to skip lines and give the appearance of choice.
- Correct versus Incorrect Syntax can drastically affect the function and output of code.
- Neat code is as important as the actual solution.
- Program documentation such as commenting is an important part of the coding process.

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

Check all that apply. 21 st 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. 21 st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication
	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		

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

- How to use conditional operators and control structures to change the flow of a program.
- How to use loops and conditional statements to solve problems.
- Identifying which type of loop is preferred for different solutions.
- Where to put curly braces to make block statements.
- How to format code so other programmers can read solutions easily.
- How to improve program efficiency through the use of control structures.

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

Denote required

This course utilizes Project Based Learning. Students will learn about various coding concepts and will have to apply their knowledge by performing various programming exercises. Projects will be assigned when the students have learned enough concepts to develop a program to complete a task.

Formative and summative evaluations in the form of programming task, 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 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

<i>Honors</i>	
Resources	
<ul style="list-style-type: none"> • 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/Technology 9-12 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 3. Methods, Classes, and Objects
Time Frame	30 days
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra	

8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.

8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.

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

Unit Overview

- Classes and Objects
- Methods and Constructors
- Various parts of methods (return type, parameters, modifiers)
- Random Class and methods of the Random Class
- Understand how methods are used to organize/consolidate code
- Using Classes/Objects to share/organize code.

Standard Number(s)

*** i.e: Math: F.LE.A.4**

i.e.: NJSLSA.R4.

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJLSA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- Why do we use methods?
- How are methods used to organize code?
- What are the purposes of having a method return a value?
- What is the void return type?
- How does a non-static method differ from a static method?
- What do parameters represent for a method?
- What does it mean to overload a method?
- What does it mean to use an object oriented program design?
- What is a class and why do we use them?
- What is the purpose of a constructor?
- What is a default constructor and when does it get invoked?
- How can classes/object be used to organize/share code?
- What is the Random class and how is it useful in simulations?

Enduring Understandings

- Methods are used to organize code into clearly defined, reusable blocks.
- Object/classes help organize code and it provides an easier way to share code.
- Understand how methods are represented and the various components of a method.
- Understand how classes are represented and the various components of a class.
- Learning to code with methods and objects is a skill that will enhance a programmer's ability to create powerful, modular and reusable code.
- Know how to incorporate the Random class and the methods of the class.
- Documentation is vitally important in terms of the development and maintenance of professional code.

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

Check all that apply. 21 st 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. 21 st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication
	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		

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

- Develop and understanding of the class structure and method structure.
- Develop the ability to create simple classes and methods.
- Know how to incorporate other classes in a program
- Realizing the important of object oriented programming and the benefits.
- Understand how classes allows collaboration on projects easier
- Incorporate the Random class in a project for simulating different events.
- Formatting of code and commenting so other programmers can read their solution easily.
- 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.
- Given method and object specifications and documentation, create the appropriate code to solve a problem in a variety of curricular areas including math, science, engineering, and English.
- Research, documentation, and communication skills needed to produce a working program.
- Develop and enhance collaboration skills.

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

Denote required

This course utilizes Project Based Learning. 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.

Teaching and Learning Activities

Activities

- Lecture and class discussion.
- Video and multimedia presentations.
- Build a console 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

Honors

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/Technology 9-12 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 4: Programming Fundamentals – Creating Programs, Syntax, Data, Variables, Input, Output
Time Frame	15 days
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra	
<p>8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.</p> <p>8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.</p> <p>9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking</p>	
Unit Overview	
<ul style="list-style-type: none"> ● Introduction to the concept of Computer Science and Computer Programming ● Introduction to Console Application Programming ● Input and Output in programming ● Variables and Data types ● Basic Mathematical Operations with numerical data types ● Basic String operations ● Algorithms ● Debugging and error checking ● Basic Problem Solving 	

Standard Number(s)*** i.e: Math: F.LE.A.4****i.e.: NJSLSA.R4.**

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJLSA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- What is computer science and computer programming?
- What are the positive and negative impacts of technology on society?
- How can we classify data and what are data types?
- How do we use variables to store various data types?
- How do we display information to the user in a console application?
- How do we receive input from a user in a console application?
- How can we use mathematical operations in our code?
- What are string operations and how are they useful?
- What is an algorithm?
- How can we create a program to solve basic problems?

Enduring Understandings

- Technology and programming languages are constantly evolving.
- Technology has impacted virtually every industry and field of endeavor in our society.
- Programs are fundamentally very specific and literal instructions with specific syntax.
- Variables and specific Data types play an important role in input/output in programs.
- There are various operations associated with the different data types.
- A program is an algorithm that is used to solve a particular problem or perform a given task.

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

<i>Check all that apply.</i> 21st Century Themes		<i>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.</i> 21st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication
	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		

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

- Understand the basic principles behind computer science and the basic ideas of computer programs.
- Understand variables and data types and apply them to simple programs
- Apply the basic concepts of input/output in console/command line programs
- Use the various operations associated with data types.
- Understand the basic concepts of problem solving and apply them to creating algorithms
- Identify several uses for computer programming in modern society.
- Understand the idea of debugging and fixing a program

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

Denote required

This course utilizes Project Based Learning. Students will learn about various coding concepts and will have to apply their knowledge by performing various programming exercises. Projects will be assigned when the students have learned enough concepts to develop a program to complete a task.

Formative and summative evaluations in the form of programming task, 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 application.
- Review and extend program functionality.
- Group and collaborative work.
- Student presentations of projects.

<p><i>Differentiation Strategies</i></p>	<ul style="list-style-type: none"> • 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 <p>Differentiation Strategies for Special Education Students Differentiation Strategies for Gifted and Talented Students Differentiation Strategies for ELL Students Differentiation Strategies for At Risk Students</p>
<p><i>Honors</i></p>	<p>N/A</p>
<p>Resources</p>	
<ul style="list-style-type: none"> • http://www.state.nj.us/education/cccs/ • http://www.corestandards.org/ELA-Literacy • http://www.state.nj.us/education/cccs/ 	

<p>Content Area/ Grade Level/ Course:</p>	<p>Mathematics 9-12 Foundations of Computer Programming: Full Year</p>
<p>Unit Plan Title:</p>	<p>Unit 5: Program Flow, Decisions, and Loops</p>

Time Frame	25 days	
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra		
<p>8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.</p> <p>8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.</p> <p>9.1 21st-Century Life & Career Skills: A Critical Thinking and Problem Solving Career and Technical Education: M. Manufacturing: Problem Solving and Critical Thinking</p>		
Unit Overview		
<ul style="list-style-type: none"> ● Program flow and changing the flow of code. ● Syntax of “if”, “if else”, “if else if”, “for”, and “while” loops. ● Boolean values, Conditional operators and compound operators. ● Use of conditionals and loops. ● Organizing code and blocking statements using correct syntax ● The need for disrupting the flow of control in programs. ● Problem solving using loops and conditionals ● Combining statements to achieve a goal. 		
Standard Number(s)	* i.e: Math: F.LE.A.4	i.e.: NJSLSA.R4.

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJSLA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- Why is it important to be able to change the flow of code?
- What is a control statement?
- What are the conditional operators and how are they incorporated in the different control structures?
- How do you use an if, if else, and if else if statements?
- How do you use a for loop and while loop?
- What are the components of a for loop?
- How can these control structures be used in solving problems/creating programs?

Enduring Understandings

- Flow of a program can be disrupted in many useful ways.
- Conditional operators play a key role in the various control structures.
- Compound operators can help reduce the lines of code and make programs more efficient.
- Programmers can use loops to eliminate lines of code and make programs more efficient.
- Programmers can use conditional statements to skip lines and give the appearance of choice.
- Correct versus Incorrect Syntax can drastically affect the function and output of code.
- Neat code is as important as the actual solution.
- Program documentation such as commenting is an important part of the coding process.

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

Check all that apply. 21 st 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. 21 st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication
	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		

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

- How to use conditional operators and control structures to change the flow of a program.
- How to use loops and conditional statements to solve problems.
- Identifying which type of loop is preferred for different solutions.
- Where to put curly braces to make block statements.
- How to format code so other programmers can read solutions easily.
- How to improve program efficiency through the use of control structures.

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

Denote required

This course utilizes Project Based Learning. Students will learn about various coding concepts and will have to apply their knowledge by performing various programming exercises. Projects will be assigned when the students have learned enough concepts to develop a program to complete a task.

Formative and summative evaluations in the form of programming task, 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 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

<i>Honors</i>	
Resources	
<ul style="list-style-type: none"> • 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/Technology 9-12 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 6. Methods, Classes, and Objects
Time Frame	25 days
Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra	

8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.

8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.

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

Unit Overview

- Classes and Objects
- Methods and Constructors
- Various parts of methods (return type, parameters, modifiers)
- Random Class and methods of the Random Class
- Understand how methods are used to organize/consolidate code
- Using Classes/Objects to share/organize code.

Standard Number(s)

*** i.e: Math: F.LE.A.4**

i.e.: NJSLSA.R4.

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJSLA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- Why do we use methods?
- How are methods used to organize code?
- What are the purposes of having a method return a value?
- What is the void return type?
- How does a non-static method differ from a static method?
- What do parameters represent for a method?
- What does it mean to overload a method?
- What does it mean to use an object oriented program design?
- What is a class and why do we use them?
- What is the purpose of a constructor?
- What is a default constructor and when does it get invoked?
- How can classes/object be used to organize/share code?
- What is the Random class and how is it useful in simulations?

Enduring Understandings

- Methods are used to organize code into clearly defined, reusable blocks.
- Object/classes help organize code and it provides an easier way to share code.
- Understand how methods are represented and the various components of a method.
- Understand how classes are represented and the various components of a class.
- Learning to code with methods and objects is a skill that will enhance a programmer's ability to create powerful, modular and reusable code.
- Know how to incorporate the Random class and the methods of the class.
- Documentation is vitally important in terms of the development and maintenance of professional code.

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

Check all that apply. 21 st 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. 21 st Century Skills
X	Global Awareness	ETA Creativity and Innovation
	Environmental Literacy	ETA Critical Thinking and Problem Solving
	Health Literacy	ETA Communication
	Civic Literacy	ETA Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy	

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

- Develop and understanding of the class structure and method structure.
- Develop the ability to create simple classes and methods.
- Know how to incorporate other classes in a program
- Realizing the important of object oriented programming and the benefits.
- Understand how classes allows collaboration on projects easier
- Incorporate the Random class in a project for simulating different events.
- Formatting of code and commenting so other programmers can read their solution easily.
- 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.
- Given method and object specifications and documentation, create the appropriate code to solve a problem in a variety of curricular areas including math, science, engineering, and English.
- Research, documentation, and communication skills needed to produce a working program.
- Develop and enhance collaboration skills.

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

Denote required

This course utilizes Project Based Learning. 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.

Teaching and Learning Activities

Activities

- Lecture and class discussion.
- Video and multimedia presentations.
- Build a console 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

Honors

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/Technology 9-11 Foundations of Computer Programming: Full Year
Unit Plan Title:	Unit 7. GUI Applications
Time Frame	40 days

Anchor Standards/Domain* *i.e: ELA: reading, writing i.e.: Math: Algebra

8.1 Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaboratively and to create and communicate knowledge.

8.2 Technology Education, Engineering, and Design: All students will develop an understanding of the nature and impact of technology, engineering, technological design, and the designed world, as they relate to the individual, global society, and the environment.

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

Unit Overview

- Difference between a console application and GUI application?
- Developing a GUI application.
- Various controls/components that are used in a GUI application
- Events in programming and how to handle various events
- Creation of a graphical program using controls and events.
- Basic Animation and collision in programming.

Standard Number(s)*** i.e: Math: F.LE.A.4****i.e.: NJSLSA.R4.**

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 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.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.
- 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.
- NJSLA.SL1. - Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- SL.11-12.4 - Present information, findings and supporting evidence clearly, concisely, and logically. The content, organization, development, and style are appropriate to task, purpose, and audience.
- 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-ETS1-4. - 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.

Intended Outcomes - {Essential Questions}

- What is the difference between a console application and GUI application?
- What is a Form/Panel and its relationship to a GUI application?
- What is a component/control and how can you use them to create a GUI?
- What are properties of the Form/Panel and Controls/Components?
- What is an event? How are they vital in a GUI application?
- How can we use events to make the keyboard and mouse do more?
- How can graphics/animation be done using a Timer control/component?
- How can we represent animation/motion in a program?
- How does collision work and how do we incorporate rate in our programs?

Enduring Understandings

- Understand the difference between a console application and GUI application.
- Know which type of problems are better represented and solved with a console vs. GUI application.
- Understand the following components: Button, Label, TextBox, RadioButtons, CheckBoxes, Timers, PictureBoxes, Progress Bars, GroupBoxes.
- Understand the properties of the Form/Panel and controls/components.
- Understand the events associated with the various components/controls.
- Understand the events associated with the keyboard and mouse.
- Understand how to accomplish animation using timers, events, and location
- Understand how collision works in programming.
- Understand the process of creating a GUI application and use it to create games and other projects.
- Documentation is vitally important in terms of the development and maintenance of professional code.

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

<i>Check all that apply.</i> 21st Century Themes		<i>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.</i> 21st Century Skills	
X	Global Awareness	ETA	Creativity and Innovation
	Environmental Literacy	ETA	Critical Thinking and Problem Solving
	Health Literacy	ETA	Communication
	Civic Literacy	ETA	Collaboration
X	Financial, Economic, Business, and Entrepreneurial Literacy		
Student Learning Targets/Objectives (Students will know/Students will understand)			
<ul style="list-style-type: none"> • Understand the difference between a console and GUI application. • Know how to create a GUI application. • Know how to use the various components/controls and understand their events. • Know how to produce animation using timers, movement, and images. • Understand the concept of collision in programming. • Know how to incorporate collisions in their programs. • Create an extensive GUI application incorporating topics learned throughout the course. • Apply communications and data analysis to the problem-solving and decision making processes in a variety of life situations. • 			
Assessments (Pre, Formative, Summative, Other) <i>common assessments with an *</i>		Denote required	

This course utilizes Project Based Learning. 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.

Teaching and Learning Activities

<i>Activities</i>	<ul style="list-style-type: none">• Lecture and class discussion.• Video and multimedia presentations.• Build a console application.• Review and extend program functionality.• Group and collaborative work.• Student presentations of projects.
<i>Differentiation Strategies</i>	<ul style="list-style-type: none">• 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
<i>Honors</i>	

Resources

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

