FREEHOLD REGIONAL HIGH SCHOOL DISTRICT

OFFICE OF CURRICULUM AND INSTRUCTION

COMPUTER SCIENCE MAGNET PROGRAM

HONORS COMPUTER SCIENCE 2

Grade Level: 10

Credits: 5

BOARD OF EDUCATION ADOPTION DATE:

AUGUST 25, 2014

FREEHOLD REGIONAL HIGH SCHOOL DISTRICT

Board of Education

Mr. Heshy Moses, President
Mrs. Jennifer Sutera, Vice President
Mr. Carl Accettola
Mr. William Bruno
Mrs. Elizabeth Canario
Mrs. Amy Fankhauser
Mrs. Kathie Lavin
Mr. Michael Messinger
Ms. Maryanne Tomazic

Mr. Charles Sampson, Superintendent
Dr. Jeffrey Moore, Director of Curriculum and Instruction

Curriculum Writing Committee

Mr. James Gill

Supervisor

Mr. Joseph Iacullo

HONORS COMPUTER SCIENCE II
COURSE PHILOSOPHY
In *Honors Computer Science II*, students will continue to develop their programming and problem solving skills. Through the creation of user-friendly programs that model and solve real world scenarios and problems, students are introduced to data structures and graphical user interfaces. A focus of this course is to create both effective and efficient programs. Additionally, students will transfer their understanding of one programming language to a variety of other languages and discover the similarities and differences in the languages.
COURSE DESCRIPTION
Students will continue to enhance their programming skills while developing their own data structures, classes, and user-friendly graphical user interface programs. They will experience and understand that, by creating their own classes, they are making the computer programming language extensible. Students will develop a graphical user interface program in Java and then create the same style programs in a graphical interface based language. They will understand how the code works behind each object, thereby increasing their abilities in computer programming.

COURSE SUMMARY	
COURSE GOALS	
CG1: The students will be able to develop their own data structures and classes in order to find efficient solutions to problems. CG2: The students will develop versatility by transferring their programming abilities to other languages. CG3: The students will be able to create advanced programs and classes that model real life applications.	
COURSE ENDURING UNDERSTANDINGS	COURSE ESSENTIAL QUESTIONS
CEU1: There are similarities and differences between all computer languages.	CEQ1: How can we transfer the skills that are acquired in one programming language to other programming languages?
CEU2: By creating original classes, you are making the computer language extensible.	CEQ2: How can we use computer programming to model and solve real world situations and problems?

UNIT GOALS & PACING		
UNIT TITLE	**UNIT GOALS**	**RECOMMENDED DURATION**
Unit 1: List Interface	Students will be able to utilize the List Interface to represent and revise data using array lists and linked lists.	1-3 weeks
Unit 2: String Manipulations	Students will be able to effectively manipulate strings to perform a variety of operations on data including mathematical functions, changing the data representation, revising the data, and analyzing data.	2 weeks
Unit 3: Java Applets	Students will be able to independently use their learning to create a Java applet and display it on the Internet.	2-3 weeks
Unit 4: Graphical User Interfaces	Students will be able to create programs that incorporate a graphical user interface.	4 weeks
Unit 5: User-defined Lists	Students will be able to implement a variety of linked lists to store and revise information.	6-8 weeks
Unit 6: Running Time and Sorting	Students will be able to create efficient recursion sorts by considering run times.	1-2 weeks
Unit 7: Introduction to a GUI Language	Students will be able to develop programs in a GUI language.	2 weeks
Unit 8: Program flow in a GUI language	Students will be able to effectively create programs that have repetition and make decisions in GUI based languages.	1-2 weeks
Unit 9: Functions in GUI	Students will be able to develop their own user-defined functions using value parameters and/or reference parameters to create programs that will perform calculations and break numbers into digits.	2-3 weeks
Unit 10: Arrays, Structures and Classes	Students will be able to represent multiple values in a single variable by using arrays, structures and developing classes in a GUI based language.	2-3 weeks
Unit 11: Graphics	Students will be able to develop programs that display basic graphics and animations.	1-2 weeks
Unit 12: Midi Applications and Files	Students will be to create programs with multiple forms that access files in order to separate different sets of input for a user on multiple pages.	2-3 weeks
Unit 13: Introduction to Python (Optional)	Students will begin to transfer their learning to other programming languages to create programs in Python that use loops, conditionals and user-defined functions to perform tasks.	2-3 weeks
Unit 14: Development for Mobile Devices (Optional)	Students will be able to create simple apps for a mobile device.	2-3 weeks

HONORS COMPUTER SCIENCE II
UNIT 1: LIST INTERFACE **SUGGESTED DURATION: 1-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to utilize the List Interface to represent and revise data using array lists and linked lists.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can create advanced programs that focus on the interaction between two lists.
3	The student can utilize the List Interface to: • perform storage, insertions, and replacements; • model real world data; • use array lists; • use linked lists.
2	The student can represent, modify and display data.
1	The student needs assistance to represent and display data using the List Interface.
0	Even with help, the student does not exhibit the ability to represent, display or modify data using the List Interface.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: An array list and linked list are different data structures that perform the same functions, though one may be best over another given the situation.	EQ1a: When two data structures from the List Interface perform an insertion, which data structure should you use?
EU2: A method from the List Interface may be effective but not efficient.	EQ2: Why should you consider run time when determining which method to use?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will write a program that uses a data structure from the List Interface to perform functions that categorize, display and manipulate data of the same type of object. Students must justify their choice in data structure. Students must also make comments about the running time of some of the methods used from the chosen data structure from the List Interface.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use of list interfaces. 1. Students will write a computer program that utilizes array lists and linked lists from Java classes to perform insertions, replacements and removals. They will get a list of objects (some containing the same data) from a user-defined class that will be added to a generic array list. The development will include: • replacing particular objects with a different object using the set command; • inserting a certain object at every even position (i.e. indices 0, 2, 4, etc.) using the add method; • removing certain objects using the remove method. The results will be demonstrated by interchangeably using “for” loops, “while” loops, enhanced “for” loops, and iterators. 2. Students will change the array list to a linked list to perform the same operations with the same code and explain that a linked list and array list perform the same functions since they implement the List Interface. They will explain how an array list and linked list are structured behind the Java code.  Students of different levels will work together. Advanced students will be assigned additional methods to develop.	Interface, Inheritance, List Interface, Array List and Linked List DOK 1	Use data structures DOK 4

HONORS COMPUTER SCIENCE II
UNIT 2: STRING MANIPULATIONS **SUGGESTED DURATION: 2 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to effectively manipulate strings to perform a variety of operations on data including mathematical functions, changing the data representation, revising the data, and analyzing data.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop advanced programs by applying binary strings.
3	The student can successfully develop programs that involve advanced string operations including using tokenizers.
2	The student can represent strings and perform the basic operations.
1	The student needs assistance to use the basic methods of the String Class.
0	Even with help, the student does not exhibit understanding of string methods.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: Strings can be manipulated in many different ways; there are situations in which these manipulations are necessary.	EQ1: When would you want to perform an operation on a string?
EU2: String variables are different than numeric variables; there are times when you need to convert a string variable to a numeric variable so as to perform mathematical operations.	EQ2: When would you want a string to become a number?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	The students will create a word frequency program. The program will separate a large string into words using a string tokenizer and analyze the frequency of each word.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will write computer programs that perform string manipulations. 1. Students will write a program that will take a string that represents a name of the form “Smith, John” and convert it to a string of the form “John Smith.” Students will use the methods charAt() and substring() to perform the operations. 2. Students will redo program 1 using a string tokenizer. 3. Students will write a program that will separate a text into words and tokenize on spaces and punctuation.  The level of expectation for individual student programs will be based on their ability.	Methods of the String Class, the Wrapper Class, String Tokenizer, Delimiters DOK 1	Apply the methods of the String Class to manipulate strings DOK 4

HONORS COMPUTER SCIENCE II
UNIT 3: JAVA APPLETS **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to independently use their learning to create a Java applet and display it on the Internet.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can convert a GUI application to an applet.
3	The student will be able to create an original applet that demonstrates understanding.
2	The student will be able to create applets by modeling other applets.
1	The student can develop a simple applet by using a previously defined applet, making minor modifications.
0	Even with help, the students are not able to develop a simple applet.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: The visual interface and interactions that applets provide can assist in communicating a message.	EQ1: Why do we need applets to communicate our information? Why is an application not enough?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	The students will use imagination and creativity to develop their own applets that involve graphics. Advanced students will create an animation of their applet.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will write Java applets that run on Applet Viewer and create a HTML file that runs the applet in Internet Explorer. 1. Students will run a Java applet that graphically displays a snowman. They will make modifications to this applet such as putting a smile on the snowman, changing the position of the arms, and adding another object. 2. Students will run their applets from the command line using the javac (Java Compiler) and show that a class file was created in Java Byte Code with a dot class extension.  Students will develop an original applet according to their interests and familiarity of the material.	Java Applet, init, Repaint, HTML DOK 1	• Develop an applet • Run the applet in Applet Viewer or Internet Explorer DOK 4

HONORS COMPUTER SCIENCE II
UNIT 4: GRAPHICAL USER INTERFACES **SUGGESTED DURATION: 4 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to create programs that incorporate a graphical user interface.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can modify the GUI to incorporate advanced features such as tool tip text methods.
3	The student will be able to create an advanced GUI that incorporates check boxes, radio buttons, and textboxes.
2	The student can use radio buttons and check boxes to complete simple programs.
1	The student can use simple dialog boxes for input and output.
0	Even with help, the student does not exhibit any understanding of using graphical user objects.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: There are private inner classes called "listeners" that wait for an event to happen.	EQ1: Why are different types of listeners used for different types of events?
EU2: There are a lot of components that are needed to create a GUI; they are designed to allow for multiple programmers.	EQ2: How are interfaces used in ways that facilitate the building of massive programs with large numbers of programmers?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will write a program that incorporates several graphical objects into one GUI program. The program will be a kiosk ordering sandwiches in an imaginary delicatessen. The program should incorporate radio buttons, labels, checkboxes and popup displays. Students will then need to write code by hand to specifications that are similar to their initial program.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop a GUI computer program that simulates an imaginary deli. Students will create code to develop radio buttons, check boxes, labels, and command buttons. JOptionPane() and tooltiptext() methods will be incorporated to enhance the display. They will incorporate listeners that are inner classes.  Students will design their own GUI of an imaginary deli according to their interests and familiarity with GUI objects.	Radio Buttons, Check Boxes, Labels, Listeners, Inner Classes DOK 1	Create a GUI that uses event listeners for user input DOK 4

HONORS COMPUTER SCIENCE II
UNIT 5: USER-DEFINED LINKED LISTS **SUGGESTED DURATION: 6-8 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to implement a variety of linked lists to store and revise information.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can merge two sorted linked lists and create a separate list while keeping the integrity of the first two lists.
3	The student can create a variety of linked lists including a sorted linked list using recursive as well as iterative methods.
2	The student can create a single linked list using methods that are recursive as well as iterative.
1	The student can create a basic single linked list with basic methods.
0	Even with help, students have no concept of creating a linked list.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: Different types of linked lists can improve the performance of certain list operations.	EQ1: Why are there different types of lists? How do you determine which to use?
EU2: Recursive methods may be more efficient than iterative methods.	EQ2: In the real world, do we differentiate between recursion and iteration? What is your proof?
EU3: An efficient recursive remove in a sorted linked list does not have to traverse the whole list.	EQ3: When is it efficient to jump out of the list early?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop a program that creates user-defined linked lists of different types, including single linked lists that are sorted and/or unsorted. The methods will emulate the Java Linked List Class. Some methods will be required to use iterative techniques and some will be required to use recursive techniques.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use of use-defined linked lists. 1. Students will create their own user-defined generic linked lists that performs the same methods as the standard Java linked list classes such as: • remove(); • remove(index); • remove(object); • add(object); • add(index, object); • set(index, object); • toString(). They will also create two recursive methods: printInOrder() and printInReverse(). They will demonstrate that their methods work correctly in a driver program given specific data from a user-defined class such as a student class. 2. Students will modify the first program by converting most of their methods to recursive methods and using a last reference (a pointer to the last element in the list). This will prepare the students for the development of a binary search tree in later courses. 3. Students will create a user-defined generic sorted linked list. They will show that only classes that implement the Comparable Interface can be used. Most methods will be recursive. They will create a remove(object) method that will leave the method once it can be determined that the object is not in the list (instead of traversing the whole list).  Advanced students will develop extra methods that merge together two sorted linked lists.	List Node, Linked List, Front Reference, Back Reference, Sorted, Efficiency DOK 1	Demonstrate the linked classes by creating a small student class DOK 4

HONORS COMPUTER SCIENCE II
UNIT 6: RUN-TIME AND SORTING **SUGGESTED DURATION: 1-2 WEEKS**

UNIT OVERVIEW			
UNIT LEARNING GOALS			
Students will be able to create efficient recursion sorts by considering run times.			
UNIT LEARNING SCALE			
4	In addition to score 3 performances, the student can compute running times of various methods that they have never seen.		
3	The student will be able to sort lists and determine the running time of the sorting techniques and various methods.		
2	The student will be able to sort lists.		
1	The student will understand basic sorting techniques.		
0	Even with help, the student does not exhibit an understanding of sorting techniques.		
ENDURING UNDERSTANDINGS		ESSENTIAL QUESTIONS	
EU1: Efficiency is an important consideration when creating a program.		EQ1: Can a program be effective and not efficient?	
EU2: The run time of a program helps determine its effectiveness.		EQ2: Can a program have a long run time and still be effective?	
NJCCCS & COMMON CORE STANDARDS			
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8			

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Write a program that will use the different sorting methods to sort primitive types and objects. The students will produce two documents: one that describes the running time of each sort, and another that traces the details of one recursive sort and one iterative sort.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will write computer programs that will test the efficiency of different sorting methods. 1. Using the code of the major sorts, students will trace the code of the sort routine using a given array. There will be a class discussion of the running time of each sort. Running time of other methods will be discussed, such as adding an element to the end of a linked list without a last reference O(n) and with a last reference O(1). 2. Students will develop a program that analyzes each sort routine by counting the number of comparisons. For each sort, 100,000 elements will be randomly generated. 3. Students will develop a sort routine for objects. They will decide which field or fields they want to sort on. They should understand that they must use classes that implement the Comparable Interface if they want to make the method generic.  The level of expectation for individual students will be based on their ability.	Bubble Sort, Quick Sort, Merge Sort And Selection Sort DOK 1	• Determine when to use comparable objects in arrays and primitive types • Create a driver that aids in the evaluation of the running time of the algorithms DOK 4

HONORS COMPUTER SCIENCE II

UNIT 7: INTRODUCTION TO A GUI LANGUAGE **SUGGESTED DURATION: 2 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to develop programs in a GUI language.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop programs that contain any of the main objects (including toolbars) to create events.
3	The student will be able to develop programs using the basic building blocks of GUI programming.
2	The student can develop programs that can create an event by clicking on a button or using a menu on a toolbar.
1	The student will be able to develop with assistance basic programs that use basic objects, variables types, and textboxes.
0	Even with help, the student does not exhibit the basic knowledge to develop a simple GUI program.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: An object that is dragged onto the canvas is created behind the program.	EQ1: How does each object in a GUI language compare to the same object created in Java?
EU2: The life and scope of different types of variables are important for efficiency in a computer program.	EQ2: Why is it important to keep the scope of a variable as narrow as possible?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop an original program involving basic GUI objects in a GUI language. The program must have all of the objects that were studied in this unit and be professional in appearance, user friendliness, and usage.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use of GUI objects in a GUI language. 1. Students will develop a GUI based program that displays the members of a band when clicking on a radio button. Radio buttons, group boxes and labels will be used. 2. Students will create a GUI based temperature conversion program (between Fahrenheit and Celsius). A program menu will allow the user to decide which conversion to use.  The level of expectation for individual students will be based on their ability.	Menu, Radio Buttons, Group Boxes, Labels DOK 1	Incorporate all of the basic objects into a GUI Visual Basic program DOK 4

HONORS COMPUTER SCIENCE II
UNIT 8: PROGRAM FLOW IN A GUI LANGUAGE **SUGGESTED DURATION: 1-2 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to effectively create programs that have repetition and make decisions in GUI based languages.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop programs that use string functions and Unicode.
3	The student will be able to develop programs using decision structures and loops.
2	The student can develop programs using decision statements, loops and string methods.
1	The student can develop programs with simple decision statements and loops.
0	Even with help, the student cannot understand how to use decision statements and loops.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1a: There is a difference between the implementation of a "for" loop and a "while" loop in a GUI based language. EU1b: Whether to use an if/else statement or a select case function is dependent on the variables and options involved. EU1c: There are times when you must change the representation of a character.	EQ1: How do you choose which structure is more efficient given the situation?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop a GUI program that incorporates decision structures within a GUI environment. Students will create a program that uses objects such as radio buttons and check boxes that will be evaluated as true or false to determine the output of a program. The program will be created to address a specific task/make a specific decision.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs in a GUI based language. 1. Students will create a GUI program that creates a sandwich order routine. A group box and two radio buttons will be used to determine if the sandwich is large or small. It will also allow the user to select sandwich toppings. This program is similar to the program written in Unit 4. The students will compare and contrast the difference between creating a GUI in Java and creating a GUI in a GUI based language. 2. Students will create a GUI program that uses a while loop to determine if a string is a palindrome.  The level of expectation for individual students will be based on their ability.	If-Then Statements, Select Case Functions, Randomize(), Messagebox(), Static Variables, Check Boxes, Input Boxes, vbTab, vbCrLf, AscW(), ChrW(), compare() DOK	• Create decision structures • Create looping structures DOK 4

HONORS COMPUTER SCIENCE II
UNIT 9: FUNCTIONS IN GUI **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to develop their own user-defined functions using value parameters and/or reference parameters to create programs that will perform calculations and break numbers into digits.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop programs that utilizes all of the business programs, mathematical functions and implement list boxes and combo boxes.
3	The student can develop programs using user-defined functions and utilizing value parameters and reference parameters.
2	The student can develop user-defined functions and demonstrate a basic understanding of the mathematical functions.
1	The student will be able to create programs that use simple user-defined functions and Visual Basic defined functions.
0	Even with help, the student cannot develop a program that uses a simple user-defined function or a Visual Basic defined function.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: There is a difference between a value parameter and a reference parameter; a programmer must decide which is best given the situation.	EQ1: How do you choose which structure is more efficient given the situation?
EU2: When working on large programs, it is best to create functions to perform tasks that may be repeated.	EQ2: Why create a function when you can just write the code in the main program?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop a program that calculates the breakdown of loan payments. The program will involve several predefined objects such as a combo box to input information and predefined business functions or user-defined functions to perform the calculations.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use of user-defined functions in a GUI based language. 1. Students will create a GUI based program that validates user input creating user-defined functions that return a Boolean value. The user-defined functions will be ValidInt(), ValidSingle(), and ValidChar(). They will validate pre-determined values for an integer, decimal and character. 2. Students will create a GUI based program that reduces fractions. 3. Students will create a die that can be rolled and which returns a value.  The level of expectation for individual students will be based on their ability.	Sub, ByVal, ByRef, Handles, Sender, Object, Function, Return, List Box, Combo Box, Present Value, Future Value DOK 1	Incorporate both user-defined functions and Visual Basic built in functions using parameters of the ByVal or ByRef type DOK 4

HONORS COMPUTER SCIENCE II
UNIT 10: ARRAYS, STRUCTURES AND CLASSES **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to represent multiple values in a single variable by using arrays, structures and developing classes in a GUI based language.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop a class inheritance hierarchy.
3	The student can develop classes that represent multiple values instead of creating many variables.
2	The student can develop programs that implement a structure array.
1	The student can set up basic arrays and define a structure with assistance.
0	Even with help, the student cannot set up an array or structure.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: A structure is the beginning of a class.	EQ1: How can we apply our understanding of a structure to representing data?
EU2: Arrays are a necessity when storing and organizing data.	EQ2: Could programming exist without arrays?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop several programs that use structures and progress to include classes. Students will start off with a structure such as a student structure that contains multiple pieces of information. They will develop this structure to a class that contains the same structure plus methods to implement functionality.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use arrays in a GUI based language. 1. Students will create a GUI based program that represents a golf game using a 2-dimensional array for 4 players and 9 holes of golf. The scores will be randomly generated and the winner of each hole will be displayed in a label. 2. Students will create a GUI based program that contains a student class consisting of a first name, last name and an unlimited number of test scores. The program will consist of two textboxes for the student's name and buttons that are used to enter the score and display the full name of the test, the minimum test grade, the maximum test grade, and the average test grade.  The level of expectation for individual students will be based on their ability.	Arrays, ReDim, Structure, Enum, New, Classes, Property, Public, ReadOnly, Inherits, Overridable, Shadows DOK 1	Use structures and classes to represent data DOK 4

HONORS COMPUTER SCIENCE II
UNIT 11: GRAPHICS **SUGGESTED DURATION: 1-2 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to develop programs that display basic graphics and animations.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop a program that displays animations.
3	The student can develop programs that use mouse events that give the user the ability to draw on the canvas.
2	The student can develop drawings and shapes such as rectangles and ellipses by using pre-defined functions.
1	The student can create shapes using basic pre-defined functions with assistance.
0	Even with help, the student cannot draw a shape on a drawing surface.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: Graphics and animation capture user attention, making a program more user-friendly.	EQ1: Can a program be effective without graphics and animation?

NJCCCS & COMMON CORE STANDARDS
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Using their imagination and creativity, students will develop an original animation program.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use animation and graphics. 1. Students will create a GUI bird flying application that simulates a bird flying. 2. Students will create a GUI scrolling marquee.  The level of expectation for individual students will be based on their ability.	Imports, Color, Graphics, SolidBrush, Point DOK 1	Develop animations in a GUI language DOK 4

HONORS COMPUTER SCIENCE II
UNIT 12: MIDI APPLICATIONS AND FILES **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will be able to create programs with multiple forms that access files in order to separate different sets of input for a user on multiple pages.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can develop programs with multiple forms that have a window menu with the ability to access each form.
3	The student can create programs with multiple forms that use dialog boxes and access files.
2	The student can develop programs that use multiple forms and access files.
1	With assistance, the student can set up a program that uses multiple forms that can access files.
0	Even with help, the student cannot create programs with multiple forms that access files.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: The use of files makes it easier to find, access, and edit data.	EQ1: How does the use of files optimize organization?
EU2: A program that incorporates elements of user interaction is easier to navigate and can be more efficient for the user.	EQ2: Why is user interaction important to a program?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Develop a multiple document program that is called “My Business Accessories.” The program will contain three forms that: • calculate mortgage payments; • determine the value of an investment; • calculate the breakdown of a paycheck.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will develop programs to practice the use creating a GUI based program with multiple forms. 1. Students will create a GUI number statistics program with a file menu that has access to several files containing lists of numbers. Once a file is chosen, the contents will be displayed in a textbox. Two buttons will display the sum and average of the numbers. 2. Students will create GUI based program with multiple forms that includes both a working calculator and a triangle calculator that will calculate the area of a triangle given all three sides. 3. Students will create an ASP.NET program that simulates an online ordering program for an imaginary pizza parlor.  The level of expectation for individual students will be based on their ability.	FileInfo, OpenFileDiaglog, StreamWriter, SaveFileDialog, Form, MainMenu, Textbox Methods DOK 1	• Use advanced features including file access systems • Create multiple form applications • Develop a simple ASP.NET application DOK 4

HONORS COMPUTER SCIENCE II
UNIT 13: INTRODUCTION TO PYTHON (OPTIONAL) **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW	
UNIT LEARNING GOALS	
Students will begin to transfer their learning to other programming languages to create programs in Python that use loops, conditionals and user-defined functions to perform tasks.	
UNIT LEARNING SCALE	
4	In addition to score 3 performances, the student can create programs in Python that use, create, and implement basic classes.
3	The student can create programs in Python that use functions and loops.
2	The student can create programs in Python that use conditionals.
1	The student can create programs in Python that use basic input and output.
0	Even with help, the student cannot create the most basic program in Python.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: There are fundamental principles to programming that are consistent regardless of the language.	EQ1: How is learning multiple programming languages similar to learning multiple spoken languages?
NJCCCS & COMMON CORE STANDARDS	
MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling. MP1, MP4, MP6, MP7, MP8	

COMMON ASSESSMENT	
ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop a program in Python that calculates the average of any number of grades using loops and user-defined functions.

SUGGESTED STRATEGIES		
ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will create a program that accesses a file of integers, sorts them, and returns the sorted list back into the file.  The level of expectation for individual students will be based on their ability.	Basics of the Python Language DOK 1	• Use functions • Use loops • Use conditions • Use input and output methods DOK 4

HONORS COMPUTER SCIENCE II
UNIT 14: DEVELOPMENT FOR MOBILE DEVICES (OPTIONAL) **SUGGESTED DURATION: 2-3 WEEKS**

UNIT OVERVIEW

UNIT LEARNING GOALS

Students will be able to create simple apps for a mobile device.

UNIT LEARNING SCALE

4	In addition to score 3 performances, the student can develop a mobile device app that queries a database and displays data.
3	The student can create a mobile device app that takes user input to create a display of output.
2	The student can develop a mobile device app that uses conditionals.
1	The student can create a simple app that displays a previously stated message.
0	Even with help, the student cannot create the most basic app for a mobile device.

ENDURING UNDERSTANDINGS	ESSENTIAL QUESTIONS
EU1: There are many qualities that make one mobile app more successful than another.	EQ1: What makes a mobile app successful?

NJCCCS & COMMON CORE STANDARDS

MA.9-12.HSN-Q.2 Define appropriate quantities for the purpose of descriptive modeling.
MP1, MP4, MP6, MP7, MP8

COMMON ASSESSMENT

ALIGNMENT	DESCRIPTION
HSN-Q.2 MP1, MP4, MP6, MP7, MP8 DOK 4	Students will develop a mobile device app that can calculate the day of the week on which the user was born.

SUGGESTED STRATEGIES

ACTIVITIES	DECLARATIVE KNOWLEDGE	PROCEDURAL KNOWLEDGE
Students will create mobile device apps. 1. Students will develop a mobile device app that displays the word “Hello.” 2. Students will develop a mobile device app that accepts an unlimited amount of test grades and outputs the average.  The level of expectation for individual students will be based on their ability.	XML File DOK 1	Develop simple apps for cell phones DOK 4