Course Title – Computer Programming II

Implement start year – 2015-2016

Revision Committee Members, email, extension –

Scot Butler sbutler@lrhsd.org ext. 8870
Chris Callinan ccallinan@lrhsd.org ext. 8364
Robert Kibler rkibler@lrhsd.org ext. 8583

Unit # 4, topic – Standard Data Structures

Transfer Goal

Students will be able to independently use their learning to construct new classes using variable types, that are either new or build upon existing classes.

Stage 1 – Desired Results

Established Goals

2009 NJCCC Standard(s), Strand(s)/CPI #
(http://www.nj.gov/education/cccs/2009/final.htm)

Common Core Curriculum Standards for Math and English
(http://www.corestandards.org/)

NJ World Class Standards
Content Area: 21st Century Learning and Careers
(http://www.state.nj.us/education/cccs/standards/9/9-4-k.htm)

9.4.12.K.66	Employ information management techniques and strategies to assist in decision-making
9.4.12.K.(3).8	Participate in a user-friendly design and development process Web-based and digital communication solution
9.4..12.K.(4).1	Identify and analyze customer software needs and requirements to guide programming and software development
9.4.12.K.(4).2	Create and use information technology strategies and

21st Century Themes
(www.21stcenturyskills.org)

_X__ Global Awareness
___Financial, Economic, Business and Entrepreneurial Literacy
___Civic Literacy
___Health Literacy
___Environmental Literacy

21st Century Skills

Learning and Innovation Skills:
_X__Creativity and Innovation
_X__Critical Thinking and Problem Solving
_X__Communication and Collaboration

Information, Media and Technology Skills:
_X__Information Literacy
___Media Literacy

projects plans when solving specific problems to deliver a product that meets customer specifications 9.4.12.K.(4).3 Identify and analyze system and software requirements to ensure maximum operating efficiency 9.4.12.K.(4).4 Demonstrate the effective use software development tools to develop software applications 9.4.12.K.(4).5 Use the software development process to design a software and deliver it to the customer 9.4.12.K.(4).6 Produce a computer application, in code, to demonstrate proficiency in developing an application using the appropriate programming language 9.4.12.K.(4).7 Implement software testing procedures to ensure qualify Products. 9.4.12.K.(4).8 Perform quality assurance tasks to produce quality	_X__ICT (Information, Communications and Technology) Literacy *Life and Career Skills:* _X__Flexibility and Adaptability _X__Initiative and Self-Direction ___Social and Cross-Cultural Skills _X__Productivity and Accountability _X__Leadership and Responsibility
Enduring Understandings: *Students will understand that . . .* EU 1 primitive data can be used to hold on to data within a program. EU 2 classes can be designed for any new type of information. EU 3 arrays are data structures that can be used to hold heterogeneous group of data.	**Essential Questions:** *EU 1* • What are the different types of primitive data? • In what ways can different primitive data types be used during program coding? • What are benefits of primitive data vs. reference type data? *EU 2* • Why is class development useful in creating efficient coding? • What are the parts necessary when designing a class? • What fields are required for a class? • What constructors are required for a class? • Will any methods be overridden or overloaded in the class design? *EU 3* • How is the size of an array determined? • When does an array become a 2-dimensional array? • When does a programmer use 2 different 1 dimensional array vs. one 2 dimensional array?

EU 4 lists are array type structures that can hold data that are not necessarily heterogeneously grouped.	*EU 4* • What are the advantages and disadvantages of using a list instead of an array? • Can lists be iterated over their entire structure, as they change their size?
Knowledge: *Students will know . . .* *EU 1* • how to declare primitive variables. • the correct primitive data type to use for a particular program. • the use of wrapper classes with primitive data. EU 2 • the declaration of fields of a given class. • how to create constructors for a program class. • the methods that can be written to allow access to the information for a designed class. EU 3 • the correct declaration and type of an array. • Array size and index ranges. • Dynamic array allocation for a one and two dimensional array. EU 4 • lists (linked and array) are dynamic array types. • access to members of a list is obtained through its member methods. • iteration through an array can be done through a structure iterator.	**Skills:** *Students will be able to . . .* *EU 1* • delcare and implement new variables, primitive in type. • convert primitive data to reference data through the wrapper classes for each primitive data type. EU 2 • design a class for an object. • create constructors for a new class. • override and overload methods that are required for a class. EU 3 • declare an array of the correct size for a program. • correctly determine the bounds of one and two dimensional arrays. • dynamically allocate data into a one and two dimensional array structure. EU 4 • create an array list that will dynamically change its size as information is added into and deleted from the list. • use a structure iterator to access each member of a list. • determine the benefits and hindrance of using a list vs. using an array.

Stage 2 – Assessment Evidence

Recommended Performance Tasks:

Other Recommended Evidence:

- Construction of a linked list to implement a structure search for a data item
- Program that uses wrapper classes to exchange data from its primitive form to its reference form
- Construction of a different class with fields, getter and setter methods, constructors, overridden methods and overloaded methods.
- Algorithms of the array and list data manipulation for a programming task.
- Quizzes on fields, constructors, classes and arrays
- Class discussion

Stage 3 – Learning Plan

Suggested Learning Activities to Include Differentiated Instruction and Interdisciplinary Connections:

- "Printer Queue" program. This program should take in jobs to be completed that were sent to a printer. The priorty for the completion of the Tasks is that of a queue. (A,M,T)
- Printing out the integers of an int using the wrapper class methods. (A,M)
- Construction of an athlete class. This class should have fields, constructors and methods and a list or array to keep track of athletes in a given sport. (A,M,T)

The following is the suggested sequence of learning activities and number of days Computer Programming II class.
(Approximate number of days: 28)

- Primitive data types (int, boolean, double, char, byte, long, short, float,)
- Wrapper classes for all primitive data types
- String data and its methods
- Class Fields
- Class Constructors
- Class Methods
- Over-riding methods
- Overloading methods
- Inheritance of data and methods
- One dimensional arrays
- Two dimensional arrays
- ArrayLists
- Linked Lists

Vocabulary

- Primitive data type
- Boolean
- Wrapper class
- Object class
- Equals method
- toString method
- HashCode method
- getClass method
- Iteration
- Dynamic Allocation
- Inheritance
- Deallociation