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 # 3 - Program Analysis

Transfer Goal –

Students will be able to independently use their learning to test, debug and check the efficiency of a program.

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 Life 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).7 Iterate through the design and development process to create a uniform Web-based or digital product

9.4.12.K.(3).8 Participate in a user-focused design and development process to produce Web-based and digital communication solution

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
_X_ ICT (Information, Communications and Technology) Literacy

9.4.12.K.(3).13 Test a digital communication product to evaluate its functionality 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 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 of 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.(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 quality products 9.4.12.K.(4).8 Perform quality assurance tasks to produce quality products. 9.4.12.K.(4).9 Perform maintenance and customer support functions to maintain software applications.	*Life and Career Skills:* _X_ Flexibility and Adaptability ___ Initiative and Self-Direction ___ Social and Cross-Cultural Skills _X_ Productivity and Accountability ___ Leadership and Responsibility
Enduring Understandings: *Students will understand that . . .* EU 1 testing and debugging programs must include boundaries and various data types	**Essential Questions:** EU 1 • What are the test data boundaries? • Will the code work with any data input?

EU 2 programs are continually tested/debugged while being developed EU 3 different types of errors can help find the source of the problem EU 4 a properly written program considers effectiveness as well as efficiency	EU 2 • How can the last written line of code be verified? EU 3 • Does the program compile and produce an acceptable solution? • Is there any input that could crash the program? EU 4 • Can a solution be accomplished with more efficient code or better memory allocation?
Knowledge: *Students will know . . .* EU 1 • that there are boundaries to an algorithm • the wrong data types can cause errors EU 2 • block programming allows for easier debugging as you code the program EU 3 • the type of error, where the error occurs and how it occurs can help to find the source of the error EU 4 • proper code considers efficiency	**Skills:** *Students will be able to . . .* EU 1 • test a program using its boundaries • limit the data type being input EU 2 • use block programming to minimize testing and debugging of the whole program EU 3 • identify the area of a program that holds the error • correct logic errors based on runtime errors EU 4 • create efficient code • debate the difference between efficient and inefficient code

Stage 2 – Assessment Evidence

Recommended Performance Tasks:

Other Recommended Evidence:

- Flow charts including potentially different paths controlled by conditional statements
- Pseudo code: handwritten outline of the program showing the order and process to be used to accomplish the task
- Algorithms: Written code of the mathematical process that will allow data to be put into an array and sorted. The mathematical process of searching for data using different techniques.
- Program maintenance. Revising a program to adjust to the needs of different data types and number of data members
- Given an algorithm identify the boundaries and what types of data are acceptable for it.
- Quizzes/Tests
- Class discussion

Stage 3 – Learning Plan

Suggested Learning Activities to Include Differentiated Instruction and Interdisciplinary Connections:

- Create a chart displaying the different types of errors, an explanation and an example of each.(A)
- By hand find the value a variable holds after so many iterations of a loop. (M,T)
- Compare a nested conditional statement to a loop structure for efficiency. (M)
- Isolate a bug in a program through the use of output statements (A, M, T)

The following is the suggested sequence of learning activities and number of days for the Computer Programming II class. (Approximate number of days 25)

- Error handling
- Develop appropriate test values to test a program
- Error categories
- Debugging techniques including output statements, step through and hand-tracing
- Pre- and post-conditions of a program
- Efficiency analysis
- Code statement execution count
- Try/Catch statements
- Throws commands
- Limitations of data types with respect to error handling and efficiency

Vocabulary

- Exception Handling
- Iterations
- Efficiency
- Run-time
- Logic Error
- Compile Error