

**Course Title:** Real World Programming

**Board Approval Date:** 4/15/13

**Credit / Hours:** 0.5credit

### **Course Description:**

Real World Programming focuses on mastery of the PA Academic Standards for Business, Computer and Information Technology, PA Academic Standards for Science and Technology, as well as PA Academic Standards for Mathematics. As students progress through Real World Programming, they will participate in a systematic study of multiple computer programming languages, programming language syntax, computer program debugging, basic design and development of computer programs, use of computer programs to solve problems, and creation of original computer programs.

Real World Computer Programming is an elective course that will explore current technological options available for programming a computer using algorithms (a specific set of instructions). While the course is designed to adapt to the evolving world of technology, it is likely a classic programming language will always be included.

Real World Computer Programming incorporates mathematical relationships, logic, creativity, and error analysis while promoting critical thinking and problem solving skills.

### **Learning Activities / Modes of Assessment:**

Large group instruction	Tests and Quizzes
Small group work	Checklists / Teacher Observation
Computer program experiments	Projects with Rubrics
Reading assignments	Writing assignments

### **Instructional Resources:**

*Python Programming Course* (Technology Cengage Learning 2010)

*More Python Programming* (Course Technology Cengage Learning 2012)

*Invent your own Computer Games with Python* (Albert Sweigart 2010)

*Wii Game Creation* (Course Technology Cengage Learning 2010)

*Android Boot Camp* (Course Technology Cengage Learning 2013)

*Programming Like a Pro for Teens, C++* (Course Technology Cengage Learning 2012)

*Learning to Programming Alice* (Pearson Education Inc. 2012)

Alice 2.2 Open Source Object Oriented Programming Language with 3D objects; alice.org

Scratch 1.4 Open Source Object Oriented Programming Language with 2D objects; scratch.mit.edu

Python 3.3 Open Source Object Oriented Programming Language with GUI interface; python.org

App Inventor Open Source Object Oriented Block Programming with GUI interface;

appinventor.mit.edu

Various instructional videos and educational websites

edmodo.com

## Course Pacing Guide

Course: **Real World Programming**

<b>Course Unit (Topic)</b>	<b>Length of Instruction (Days/Periods)</b>
1. Introduction to Real World Programming	20 days
2. Real World Programming Applications	50 days
3. Analysis of Real World Program Design and Development	<u>10 days</u>
<b>DAYS TOTAL</b>	<b>80 Days</b>

Topic: 1. Introduction to Real World Programming

Days: 20

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:	Understand:	Do:
<p>A program is a series of commands performed by a computer.</p> <p>Several different programming languages exist.</p> <p>Computer programs use algorithms. Algorithms are a finite set of steps that provide a general solution to a problem.</p> <p>Various computer programming languages use different types of source code with different strengths and weaknesses.</p>	<p>Knowing how to read a computer program and translate it into English is an essential programming practice.</p>	<p><b>15.4.12.H. – Unranked</b> Use programming languages to develop logical thinking and problem solving skills.</p> <p><b>15.4.8.I. – Unranked</b> Solve a problem with an algorithm.</p> <p><b>15.4.12.J. – Unranked</b> Create a complex computer program to solve a problem.</p> <p><b>15.4.12.I. – Unranked</b> Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p><b>2.2.8.C – Essential</b> EVALUATE NUMERICAL EXPRESSIONS - Use the order of operations to evaluate numerical expressions.</p> <p><b>2.4.8.B – Important</b> CONNECTIONS - Use if...then statements to construct simple valid arguments.</p> <p><b>2.5.8.A – Important</b> PROBLEM SOLVING - Develop a plan to analyze a problem, identify the information needed to solve the problem, carry out the plan, apply estimation skills as appropriate, check whether the plan makes sense, and explain how the problem was solved in grade appropriate contexts.</p> <p>3.4.10.A2. Interpret how systems thinking applies logic and creativity with appropriate comprises in complex real-life problems.</p> <p>CC.3.5.11-12.E. Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.</p> <p>CC.3.5.11-12.D.</p>

Topic: 1. Introduction to Real World Programming

Days: 20

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:

Understand:

Do:

Determine the meaning of symbols, key terms and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 11-12 text topics.

Topic: 2. Real World Programming Applications

Days: 50

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:	Understand:	Do:
<p>Know how to write a computer program using different computer programming languages.</p> <p>Know how to solve problems using various programming languages.</p> <p>Know how to use appropriate syntax based on the computer program being utilized.</p> <p>Know how to analyze the computer program for errors and how to fix them.</p> <p>Know how to use comments within the program code to document how program works.</p>	<p>Designing, writing, implementing and testing computer programs are essential programming skills.</p>	<p><b>15.4.12.H. – Unranked</b> Use programming languages to develop logical thinking and problem solving skills.</p> <p><b>15.4.12.I. – Unranked</b> Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p><b>15.4.12.A. – Unranked</b> Apply the creative and productive use of emerging technologies for educational and personal success.</p> <p><b>15.4.12.J. – Unranked</b> Create a complex computer program to solve a problem.</p> <p><b>2.2.8.C – Essential</b> EVALUATE NUMERICAL EXPRESSIONS - Use the order of operations to evaluate numerical expressions.</p> <p><b>2.4.8.B – Important</b> CONNECTIONS - Use if...then statements to construct simple valid arguments.</p> <p><b>2.5.8.A – Important</b> PROBLEM SOLVING - Develop a plan to analyze a problem, identify the information needed to solve the problem, carry out the plan, apply estimation skills as appropriate, check whether the plan makes sense, and explain how the problem was solved in grade appropriate contexts.</p> <p>3.4.10.D2. Diagnose a malfunctioning system and use tools, materials and knowledge to repair it.</p> <p>3.4.10.C1. Apply the components of the technological design process.</p> <p>3.4.10.C2</p>

Topic: 2. Real World Programming Applications

Days: 50

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:	Understand:	Do:
		<p>Analyze a prototype and/or create a working model to test a design concept by making actual observations and necessary adjustments.</p> <p>CC.3.5.11-12.E.</p> <p>Analyze how the text structures information or ideas into categories or hierarchies, demonstrating understanding of the information or ideas.</p> <p>CC.3.5.11-12.D</p> <p>Determine the meaning of symbols, key terms and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 11-12 texts and topics.</p> <p>CC.3.5.9-10.G</p> <p>Translate quantitative or technical information expressed in words in a text into visual form and translate information expressed visually or mathematically into words.</p>

Topic: 3. Analysis of Real World Program Design and Development

Days: 10

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:	Understand:	Do:
<p>How to interpret various blocks of code from multiple computer programs.</p> <p>How to analyze the strengths and weaknesses of various blocks of code from multiple computer programs.</p> <p>How to determine what programming language or application to use to solve a specific problem.</p> <p>How to demonstrate use of a computer programming language to solve a specific problem.</p>	<p>Multiple computer programming languages can be used to write code that will perform a similar task, each with different strengths and weaknesses.</p>	<p><b>15.4.12.J. – Unranked</b> Create a complex computer program to solve a problem.</p> <p><b>15.4.12.I. – Unranked</b> Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p><b>15.4.12.H. – Unranked</b> Use programming languages to develop logical thinking and problem solving skills.</p> <p><b>2.2.8.C – Essential</b> EVALUATE NUMERICAL EXPRESSIONS - Use the order of operations to evaluate numerical expressions.</p> <p><b>2.4.8.B – Important</b> CONNECTIONS - Use if...then statements to construct simple valid arguments.</p> <p><b>2.5.8.A – Important</b> PROBLEM SOLVING - Develop a plan to analyze a problem, identify the information needed to solve the problem, carry out the plan, apply estimation skills as appropriate, check whether the plan makes sense, and explain how the problem was solved in grade appropriate contexts.</p> <p>3.4.10.C1. Apply the components of the technological design process.</p> <p>3.4.10.C2. Analyze a prototype and/or create a working model to test a design concept by making actual observations and necessary adjustments.</p> <p>3.4.10.A2. Interpret how systems thinking applies logic and creativity with appropriate comprises in complex real-life problems.</p>

Topic: 3. Analysis of Real World Program Design and Development

Days: 10

Subject(s): Technology

Grade(s): 9th, 10th, 11th, 12th

Know:	Understand:	Do:
		<p>CC.3.5.9-10.G.</p> <p>Translate quantitative or technical information expressed in words in a text into visual form and translate information expressed visually or mathematically into words.</p> <p>CC.3.5.11-12.B.</p> <p>Write informative or explanatory texts, including the narration of historical events, scientific procedures, or experiments or technical process.</p>