Southern York County School District Instructional Plan

Name:	**Dates: August–September**
Course/Subject: Computer Programming	**Unit Plans: 1 and 2**

Stage 1 – Desired Results

PA State Standard(s) Addressed:

2.2.11.D **Describe and explain the amount of error that may exist in a computation using estimates.**

2.2.11.E **Recognize that the degree of precision needed in calculating a number depends on how the results will be used and the instruments used to generate the measure.**

2.4.11.B **Construct valid arguments from stated facts.**

2.4.11.C **Determine the validity of an argument.**

Understanding(s):
Students will understand that…

1. Computer Science is integrated into many disciplines.
2. C++ is an AP programming language because it has widespread use in industry and academia and because it supports object-oriented programming.
3. Programmers modify software to add new services to a computer, but rarely modify hardware.
4. Most computer errors are due to errors in software introduced by the programmer, not in the hardware.
5. The software development life cycle serves as an illustration of the requirements of good programming.
6. Most programming activity should occur during the analysis and design of a program, rather than during the coding in a particular programming language.
7. Object-oriented programming is an accepted technique for developing good software systems.
8. All C++ programs have a similar structure.

Essential Question(s):

- To what extent does the study of computer science develop problem solving skills?
- How can computer programming be used in real world applications?
- How are solutions determined?

Learning Objectives: ***Students will know…***	***Students will be able to…***
▪ The history of computing. ▪ The difference between hardware and software. ▪ The difference between low-level programming languages and high-level languages. ▪ The history of the C++ programming language and its importance in industry ▪ System software, such as the computer's operating system (Windows XP) and the services it provides. ▪ Computer Science ▪ Computer Architecture ▪ Computer Languages ▪ Program Development Cycle ▪ Top-Down Design ▪ Data Types and Output	▪ Define what Computer Science is and is not. ▪ Create a timeline for the history of computers. ▪ List hardware and software components. ▪ Write whether a computer language is high-level or low-level. ▪ Explain the usefulness of the programming environment's project manager, editor, compiler, and run-time system. ▪ Describe the six phases of the software system life cycle.. ▪ Define what an algorithm is and how it is developed using top-down design and stepwise refinement. ▪ List C++ reserved words and library identifiers. ▪ List the basic components of a C++ program. ▪ Identify and use the data types int, double, and char. ▪ Identify the floating point form and fixed form of a real number. ▪ Write code to produce output on the screen.
Name:	**Dates: September**
Course/Subject: Computer Programming	**Unit Plan: 3**

Stage 1 – Desired Results

PA State Standard(s) Addressed:

- **2.1.11.A Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms).**
- **2.2.11.A Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
- **2.4.11.B Construct valid arguments from stated facts.**
- **2.4.11.C Determine the validity of an argument.**
- **2.4.11.E Demonstrate mathematical solutions to problems.**
- **2.5.11.A Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
- **2.5.11.B Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**
- **2.5.11.C Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
- **2.5.11.D Conclude a solution process with a summary of results and evaluate the**

degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.	
Understanding(s): ***Students will understand that…*** 1. Entering data and performing calculations are the power behind a computer. 2. The choice of numeric data type is determined by the requirements of the program. 3. Computers can represent and store numerous types of information. 4. Arithmetic operators are ranked by precedence. 5. Mixed-mode arithmetic produces standard results. 6. Names of variables should reflect their purpose and be initialized before use.	**Essential Question(s):** ▪ To what extent is programming the best option to solve a problem?
Learning Objectives: ***Students will know…*** ▪ There are three different kinds of program errors – syntax, run-time, and logic. ▪ Computer security issues, such as privacy, intrusive hacking, and viruses. Data Types ▪ Mixed-mode expressions ▪ Memory usage for data ▪ Keyboard data entry ▪ Library function usage	 ***Students will be able to…*** ▪ Use numerical data and their operators to perform calculations. ▪ Convert mixed-mode operations to a particular data type. ▪ Describe memory storage of data. ▪ Use the standard input stream to enter data from the keyboard. ▪ Declare and use string variables. ▪ Use the ASCII code to represent character data. ▪ Use pre-defined library functions in programs.
Name:	**Dates: October**
Course/Subject: Computer Programming	**Unit Plan: 4**

Stage 1 – Desired Results

PA State Standard(s) Addressed:

2.1.11.A	**Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms).**
2.2.11.A	**Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
2.4.11.B	**Construct valid arguments from stated facts.**
2.4.11.C	**Determine the validity of an argument.**
2.4.11.E	**Demonstrate mathematical solutions to problems.**
2.5.11.A	**Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
2.5.11.B	**Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**

2.5.11.C	**Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
2.5.11.D	**Conclude a solution process with a summary of results and evaluate the degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.**

Understanding(s):
Students will understand that…

1. The three major components of a function are the function declaration, the function call, and the function body with the header.
2. Only one value may be returned in a function using a return statement, where as multiply values may be returned when using parameter.
3. The values obtained by the value parameters are given to the functions.
4. Reference parameters look at the same memory location as its actual parameters.
5. Constant reference parameters are used to conserve memory, but not allow changes to the data structure.
6. Modular programming implements top-down design and bottom-up testing.

Essential Question(s):

- How is top-down design used in problem solving and where do you see it used in everyday events/

Learning Objectives:
Students will know…

- The difference between value and reference parameters.
- When it is appropriate to use local and global identifiers
- Abstract data types
- Block-structured
- Cohesive subprogram
- Functional abstraction
- Function applications
- Function-oriented programming
- Global identifier
- Library header file
- Library implementation file
- Local identifier
- Locally declared data
- Manifest interface
- Modularity
- Overloaded functions
- Procedural abstraction
- Scope of an identifier
- Structure theorem
- Stub programming
- Subblock

Students will be able to…

- Use and explain the process of bottom-up testing and modularity.
- Employ the use of structured programming.
- Create user-defined functions.
- Use correct syntax when declaring and using a function.
- Use stubs and drivers to test program flow.
- Use parameters in functions.
- Distinguish between value and reference parameters.
- Use functions to design programs.
- Create a library of functions.
- Determine the scope of an identifier.
- Control side effects in a program.

Name:	**Dates: November—December**

Course/Subject: Computer Programming	Unit Plan: 5

Stage 1 – Desired Results

PA State Standard(s) Addressed:

- **2.1.11.A Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms**
- **2.2.11.A Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
- **2.4.11.B Construct valid arguments from stated facts.**
- **2.4.11.C Determine the validity of an argument.**
- **2.4.11.E Demonstrate mathematical solutions to problems.**
- **2.5.11.A Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
- **2.5.11.B Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**
- **2.5.11.C Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
- **2.5.11.D Conclude a solution process with a summary of results and evaluate the degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.**

Understanding(s):
Students will understand that…

1. By using expressions that evaluate to be true or false enable programs to be written based upon conditions.

Essential Question(s):

- Why is it important to be able to use conditional statements?
- How do conditional statements expand your ability to write programs?

Learning Objectives:
Students will know…

- The nature and use of a Boolean data type in C++.
- The difference between = and ==.
- The appropriate ways to prioritize expressions and operations.
- When it is advantageous to use a switch over an if statement and vice versa.
- Assertion
- Compound Boolean expression
- Compound statement
- Extended if statement
- Logical operators
- Negation
- Nested if statement
- Program proof
- Relational operator
- Robust
- Selection statement
- Short-circuit Evaluation
- Simple Boolean expression

Students will be able to…

- Construct and evaluate Boolean expressions.
- Implement selection statements to make decisions.
- Design and test if statements.
- Design and test if…else statements.
- Design and test switch statements.

Name:	Dates: January–February
Course/Subject: Computer Programming	Unit Plan: 6

Stage 1 – Desired Results

PA State Standard(s) Addressed:

- **2.1.11.A** **Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms).**
- **2.2.11.A** **Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
- **2.4.11.B** **Construct valid arguments from stated facts.**
- **2.4.11.C** **Determine the validity of an argument.**
- **2.4.11.E** **Demonstrate mathematical solutions to problems.**
- **2.5.11.A** **Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
- **2.5.11.B** **Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**
- **2.5.11.C** **Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
- **2.5.11.D** **Conclude a solution process with a summary of results and evaluate the degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.**

Understanding(s):
Students will understand that…

1. The increment (or decrement) of the loop control variable is done explicitly.
2. A looping structure is used when a task is to be performed numerous times.

Essential Question(s):

- Why is it important to repeat a sequence of instructions?
- How are loops exemplified in our daily lives?

Learning Objectives:
Students will know…

- The difference between a pretest loop and a posttest loop.
- The difference between a fixed repetition loop and a variable condition loop.
- When it is appropriate to use a *for* loop, a *while* loop, or a *do-while* loop for a given problem.
- The major classifications of loops: Pre-test versus post-test, and fixed repetition versus variable repetition.
- Accumulator
- Control Variable
- Counter
- Data validation
- Decrement
- Fixed Repetition (iterated) loop
- Infinite loop
- Input assertion

Students will be able to…

- Explain the difference between a pretest and post-test loop.
- Identify and use a fixed repetition and variable condition loop.
- Identify the loop control variable.
- Explain the flow of control of a fixed repetition loop.
- Use a *for* loop that counts up or down.
- Explain the flow of control of a variable condition loop.
- Identify a *while* and *do-while* loop as a variable condition loop.
- Identify and use a *while* loop as a pretest loop.
- Identify and use a *do-while* loop as a post-test loop.
- Explain how input/output assertions and variants/invariants can be used to verify loops.

▪ Loop invariant ▪ Loop variant ▪ Nested loop ▪ Output assertion ▪ Post-test (exit-controlled) loop ▪ Pretest condition ▪ Pretest (entranced-controlled) loop ▪ Sentinel value ▪ Variable condition loop	▪ Identify and use nested loops. ▪ Use a selection statement within the body of a loop. ▪ Use a loop within an option of a selection statement.
Name:	**Dates: March**
Course/Subject: Computer Programming	**Unit Plan: 7**

Stage 1 – Desired Results

PA State Standard(s) Addressed:

- **2.1.11.A Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms).**
- **2.2.11.A Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
- **2.4.11.B Construct valid arguments from stated facts.**
- **2.4.11.C Determine the validity of an argument.**
- **2.4.11.E Demonstrate mathematical solutions to problems.**
- **2.5.11.A Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
- **2.5.11.B Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**
- **2.5.11.C Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
- **2.5.11.D Conclude a solution process with a summary of results and evaluate the degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.**

Understanding(s): ***Students will understand that…*** 1. Files allow the semi-permanent storage of data. 2. Most file input algorithms assume detailed awareness about the format of the data in the file. 3. Obtaining a file name from a function adds a great deal of flexibility to a program.	**Essential Question(s):** ▪ Why is it important to be able to store data on a storage device? ▪ Where is data stored and for what purpose?

Learning Objectives: ***Students will know…***	***Students will be able to…***
▪ Buffer ▪ Destination device ▪ End of input stream ▪ Input file stream ▪ Output file stream ▪ Serial processing ▪ Source device ▪ Stream	▪ Use file streams to obtain input and output data. ▪ Use loops with file streams. ▪ Design functions for use with file streams. ▪ Use file streams with strings. ▪ Distinguish character-level input and output from the input and output of other data types.
Name: Ryan Leiphart	**Dates: April—May**
Course/Subject: Computer Programming	**Unit Plan: 8**

Stage 1 – Desired Results

PA State Standard(s) Addressed:

- **2.1.11.A Use operations (e.g., opposite, reciprocal, absolute value, raising to a power, finding roots, finding logarithms).**
- **2.2.11.A Develop and use computation concepts, operations and procedures with real numbers in problem-solving situations.**
- **2.4.11.B Construct valid arguments from stated facts.**
- **2.4.11.C Determine the validity of an argument.**
- **2.4.11.E Demonstrate mathematical solutions to problems.**
- **2.5.11.A Select and use appropriate mathematical concepts and techniques from different areas of mathematics and apply them to solving non-routine and multi-step problems.**
- **2.5.11.B Use symbols, mathematical terminology, standard notation, mathematical rules, graphing and other types of mathematical representations to communicate observations, predictions, concepts, procedures, generalizations, ideas and results.**
- **2.5.11.C Present mathematical procedures and results clearly, systematically, succinctly and correctly.**
- **2.5.11.D Conclude a solution process with a summary of results and evaluate the degree to which the results obtained represent an acceptable response to the initial problem and why the reasoning is valid.**

Understanding(s): ***Students will understand that…***	**Essential Question(s):**
1. Situations exist where a program manipulates large amounts of similar data. 2. Array operations such as sorting, searching, deletions, and insertions are necessary to work with large amounts of data. 3. A vector contains items of the same type, which are located with an index position. 4. A vector contains a fixed number of storage locations for its items.	▪ How do arrays enable the programmer to deal with large amounts of data? ▪ Why do arrays decrease the workload of the programmer? ▪ Why is it important to be able to sort or search a list of data?

5. Loops are used to sum the values of vector components, find the maximum or minimum value in a vector, find the index of the maximum or minimum value, and to determine the number of cells in a given vector. 6. The difference between a vector's physical size (the number of cells allocated for storage) and its logical size (the number of data items currently stored in these cells). 7. Vectors are passed as parameters and returned as the values of functions. 8. Constant reference should be used when a function does not change the data in a vector parameter.	