

## Wilson Area School District Planned Course Guide

**Title of planned course:** Introduction to Computer Programming

**Subject Area:** Business

**Grade Level:** 9-12

**Course Description:** This introductory programming course, using Alice 2.2, is designed to introduce students to basic programming. Alice uses the modern approach to computer programming known as OOP or Object Oriented Programming. A “hands-on” approach will be used to allow students to practice the principles of the Alice language.

**Time/Credit for this Course:** 0.5 Academic Year / 0.5 Credits

**Curriculum Writing Committee:** Kari Maskalis

## Curriculum Map

**Unit 1:** Introduction to Alice

**Unit 2:** Program Design and Implementation

**Unit 3:** Built-in Functions and Expression

**Unit 4:** Classes, Objects, Methods and Parameters

**Unit 5:** Events and Event Handling

**Unit 6:** Functions and If/Else

**Unit 7:** Definite and Conditional Loops

**Unit 8:** Recursion

**Unit 9:** Lists and List Processing

**Unit 10:** Variables and Inheritance

**Wilson Area School District  
Planned Course Materials**

**Course Title:** Introduction to Computer Programming

**Teacher Resources:** [www.alice.org](http://www.alice.org)  
[www.aliceprogramming.net](http://www.aliceprogramming.net)  
<http://www.cs.duke.edu/csed/alice/aliceInSchools/>  
<http://www.dickbaldwin.com/tocalice.htm>  
<http://www.alice.org/community/>  
[www.bluepelicanjava.com](http://www.bluepelicanjava.com)

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Introduction to Programming

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Define computer programming
- Summarize the history of computer programming
- Identify current challenges within the computer programming field
- Understand the benefits of using Alice.

**Core Activities:** Students will complete/participate in the following:

- Alice Tutorial #1
- Alice Tutorial #2
- Alice Tutorial #3
- Alice Tutorial #4
- The Essentials of Alice (Bunny)
- The Essentials of Alice (Kangaroo)
- Shark Attack
- Getting Started Space

**Extensions:**

- Fish Fairy
- Princess and Dragon

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Program Design and Implementation

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Demonstrate proper use of control statements
- Select arguments that support the intended program actions
- Export code for printing.

**Core Activities:** Students will complete/participate in the following:

- Skater World
- Outdoor Adventure
- How Tall Are You?
- Decisions and Functions Assessment

**Extensions:**

- Working with Numbers
- Properties and Functions
- Transformations Challenge
- Headshots

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Built-in Functions and Expression

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Explain the relationship between method calls and arguments
- Write their own methods
- Modify the properties of an object
- Create and rename new objects from predefined classes.

**Core Activities:** Students will complete/participate in the following:

- Kangaroo Visits Friend
- Parameter's Assessment
- Creating Simple Objects
- Texture Map Assessment
- Making Objects Move in Unison
- Lists Assessment
- Sharing Alice

**Extensions:**

- Changing Color Advanced Functions
- Random Numbers Assessment
- Making Folders in the Local Gallery

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects



## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Classes, Objects, Methods and Parameters

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Explain the relationship between method calls and arguments
- Use variable in expressions to calculate new results
- Understand the difference between methods and functions
- Implement visual effects and animation.

**Core Activities:** Students will complete/participate in the following:

- Scene Changes using Environment
- Scene Changes Using Lens Cap
- Bunny Eat Broccoli
- Simple Loops Assessment
- Lights Camera Action
- BDE Assessment

**Extensions:**

- Repetition Everywhere
- Repetition Assessment
- Simple Quiz
- Simple Quiz Assessment

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Events and Event Handling

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Break down a program into smaller pieces by using objects and writing their own methods
- Explain the purpose of using parameters in writing programs
- Understand the need for writing your own function
- Write class-level functions.

**Core Activities:** Students will complete/participate in the following:

- Changing Camera Views
- Checking for Collisions
- Click on Object to Answer
- Quiz Using Ask User
- Click on the Match

**Extensions:**

- Visual Lists
- Shark vs. TRex
- Making Billboards
- Let's Play Catch
- What Can We Burn

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Functions and If/Else

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Identify the events that fire once and those that fire repeatedly
- Create events to respond to user input
- Understand the BDE event
- Explain how events make programs interactive
- Implement events to build sophisticated program code.

**Core Activities:** Students will complete/participate in the following:

- Restricting Events
- Animating Objects
- Playing Music
- Creating Poses
- Historical Tour
- Book Report

**Extensions:**

- Alice Buys a Scooter
- Stopwatch
- Countdown Timer
- Scorekeeper Object

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Definite and Conditional Loops

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Define events
- Use events to monitor conditions and variable changes
- Use relational operators for condition based decisions
- Determine Boolean results.

**Core Activities:** Students will complete/participate in the following:

- Fader Objects
- Scene Changes
- Visual Arrays
- Nonvisual Arrays
- Coloring Randomly
- Timer
- Simple Collision
- Advanced Collision

**Extensions:**

- Harry Potter Challenge
- Fun with Squares
- Tic-Tac-Toe Challenge
- Trigonometry Prom Challenge
- Calculator Game Challenge

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects



## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Recursion

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Demonstrate knowledge of logical operators to create complex situations
- Create nested if/else statements and loops
- Utilize while and loop statements
- Use control loop variable
- Recognize proper decision control statements
- Describe how to use recursion in a program.

**Core Activities:** Students will complete/participate in the following:

- Distance Challenge
- Boat Racing Challenge
- Asteroids

**Extensions:**

- Building the solar System
- Let's Race Keyboarding Game
- Alice Pong

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects

## Curriculum Scope & Sequence

**Planned Course:** Introduction to Computer Programming

**Unit:** Lists and List Processing

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Create lists and arrays
- Modify lists and arrays
- Compare and contrast lists and arrays
- Use an array visualization object.

**Core Activities:** Students will complete/participate in the following:

- Create a Boat Racing Game
- Treasure Hunt
- Piñata Game
- Wizard Game

**Extensions:**

- Cooking Show
- Line Up
- Bunny Eats Broccoli Exercise
- Coloring Randomly Exercise

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- Formative:

- Multi-level questioning
- Discussion, class assignments
- **Summative:**
  - Projects

## **Curriculum Scope & Sequence**

**Planned Course:** Introduction to Computer Programming

**Unit:** Variable and Inheritance

**Time frame:** 1-2 weeks

**State Standards:** 5.4.12.D, G-K

**Essential content/objectives:** At end of the unit, students will be able to:

- Develop list methods and functions
- Access elements stored in lists and arrays
- Change the contents of lists
- Randomly select an object in a list or array.

**Core Activities:** Students will complete/participate in the following:

- Timer Exercise
- Wizard Game Exercise
- Line Up Exercise
- Alice Final Project

**Extensions:**

- Free Program

**Remediation:**

- Tutoring
- Resubmission of previous programmed worlds

**Instructional Methods:**

- Direct instruction
- Guided and independent practice
- Projects

**Materials & Resources:**

- Computer, Internet, and Projector
- SMART Board
- Alice software

**Assessments:**

- **Formative:**
  - Observation
  - Multi-level questioning
  - Discussion, class assignments
- **Summative:**
  - Projects