Coding in Minecraft Expert Coding in Minecraft with JavaScript

Scope and Sequence

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
Lesson 1: Variables	Learners will understand the definition of variables. Learners will understand the different types of variables and when they should be used in code. Learners will be able to create variables and assign them values. Learners will understand the definition of computing and computing networks.	3B-AP-12: Compare and contrast fundamental data structures and their uses. 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible. 3B-AP-10: Use and adapt classic algorithms to solve computational problems.	CSN-1.A: Explain how computing devices work together in a network AAP-1.A: Represent a value with a variable AAP-1.B: Determine the value of a variable because of an assignment. CSN-1.A: Explain how computing devices work together in a network	5 hrs

© Prodigy Learning | prodigylearning.com | codingcredentials.com

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3B-CS-02: Illustrate ways computing systems implement logic, input, and output through hardware components.		
Lesson 2: Operators	Learners will understand operators and how they are used in code. Learners will be able to identify and use operators in JavaScript. Learners will be able to evaluate expressions that use arithmetic operators.	3B-AP-12: Compare and contrast fundamental data structures and their uses.	AAP-2.E: For relationships between two variables, expressions, or values: a. Write expressions using relational operators. b. Evaluate expressions that use relational operators.	5 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
	Learners will be able to write expressions using relational operators. Learners will be able to evaluate expressions that use relational operators.			
Lesson 3:	Learners will understand what a conditional is and why it is used in code. Learners will learn what an If statement is and how to use it in their code. Learners will learn what an If / Else statement is and how to use it in their code.	3A-NI-04: Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing. 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.	AAP-2.G: Express an algorithm that uses selection without using a programming language. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements.	8 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3A-AP-15: Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible. 3B-AP-10: Use and adapt classic algorithms to solve computational problems.		

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
Lesson 4: Advanced Logic	Learners will learn what an If / Else If / Else statement is and why it is used in code. Learners will learn how to use an If / Else If / Else statement in their code. Learners will learn what nested conditionals are and why they are used in code. Learners will learn how to use nest conditionals in their code. Learners will learn what the different types of computing are including their strengths, weaknesses, and when to use them.	3A-CS-02: Compare levels of abstraction and interactions between application software, system software, and hardware layers 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-15: Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	AAP-2.G: Express an algorithm that uses selection without using a programming language. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements. AAP-2.I: For nested selection: a. Write nested conditional statements. b. Determine the result of nested conditional statements.	8 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible. 3B-AP-10: Use and adapt classic algorithms to solve computational problems.		
Lesson 5: Loops	Learners will be able to identify and build code which uses while loop statements. Learners will be able to identify and build code which uses for loop statements. Learners will be able to identify and build code which uses nested loops.	3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	AAP-2.K For iteration: a. Write iteration statements. b. Determine the result or side effect of iteration statements.	8 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
Lesson 6: Coding Practice	Learners will be able to identify and build code that uses variables. Learners will be able to identify and use operators in code. Learners will be able to identify and build code using complex selections and conditionals. Learners will be able to identify and build loops and nested loops that use complex compound conditionals.	3B-AP-10: Use and adapt classic algorithms to solve computational problems. 3B-AP-12: Identify and Use Operators in JavaScript 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	AAP-1.A: Represent a value with a variable AAP-2.E: For relationships between two variables, expressions, or values: a. Write expressions using relational operators. b. Evaluate expressions that use relational operators. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements AAP-2.I: For nested selection: a. Write nested conditional statements. b. Determine the result of nested conditional statements. AAP-2.K For iteration: a. Write iteration statements. b. Determine the	8 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
			result or side effect of iteration statements.	
Lesson 7:	Learners will build a game using all the programming knowledge previously learned such as variables, operators, logic, and loops.	3B-AP-12: Identify and Use Operators in JavaScript 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program	AAP-2.E: For relationships between two variables, expressions, or values: a. Write expressions using relational operators. b. Evaluate expressions that use relational operators. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements AAP-2.I: For nested selection: a. Write nested conditional statements. b. Determine the result of nested conditional statements.	10 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		performance, and explain the benefits and drawbacks of choices made.	AAP-2.K For iteration: a. Write iteration statements. b. Determine the result or side effect of iteration statements.	
Lesson 8: Functions	Learners will be able to understand the concept of modularity and identify when modularity can or should be used in code Learners will be able to identify, understand, call, and build functions Learners will be able to identify, understand, call and	3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3B-AP-15: Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution. 3A-AP-16: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.	AAP-3.B: Explain how the use of procedural abstraction manages complexity in a program. CRD-2.B: Explain how a program or code segment functions. AAP-3.A: For procedure calls: a. Write statements to call procedures. 3.B b. Determine the result or effect of a procedure call	7 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
	build functions that take parameters or return data. Learners will be able to identify and understand algorithmic efficiency.	3A-AP-18: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. 3B-AP-14: Construct solutions to problems using student-created components, such as procedures, modules and/or objects.	AAP-3.C: Develop procedural abstractions to manage complexity in a program by writing procedures.	
Lesson 9: Algorithms	Learners will compare multiple algorithms to determine if they yield the same side effect or result. Learners will create algorithms. Learners will combine and modify existing algorithms.	3B-AP-11: Evaluate algorithms in terms of their efficiency, correctness, and clarity. 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects 3B-AP-10: Use and adapt classic algorithms to solve computational problems.	AAP-2.L: Compare multiple algorithms to determine if they yield the same side effect or result AAP-2.M: For algorithms: a. Create algorithms. B. Combine and modify exisiting algorithms.	7 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3B-AP-22: Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).		
Lesson 10: Collections	Learners will be able to identify, understand when to use and build code which uses lists (creation, indexing, modifying, traversing, common methods)	3A-AP-14: Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible.	AAP-1.C: Represent a list or string using a variable. 3.A AAP-1.D For data abstraction: a. Develop data abstraction using lists to store multiple elements 3.B b. Explain how the use of data abstraction manages complexity in program code. 3.C AAP-2.N For list operations: a. Write expressions that use list indexing and	7 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
			list procedures. 2.B b. Evaluate expressions that use list indexing and list procedures. 4.B	
Lesson 11: Structure and Error Handling	Learners will be able to document code using comments and documentation stings. Learners will be able to correctly use indentation to structure code appropriately. Learners will describe the purpose of a code segment or program by writing documentation.	3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible. 3A-AP-23: Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. 3B-AP-20: Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (code documentation) in a group software project.	CRD-2.G: Describe the purpose of a code segment or program by writing documentation.	7 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
Lesson 12: Programming Practice	Learners will be able to identify and build code that uses variables. Learners will be able to identify and use operators in code. Learners will be able to identify and build code using complex selections and conditionals. Learners will be able to identify and build loops and nested loops that use complex compound conditionals.	3B-AP-10: Use and adapt classic algorithms to solve computational problems. 3B-AP-12: Identify and Use Operators in JavaScript 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	AAP-1.A: Represent a value with a variable AAP-2.E: For relationships between two variables, expressions, or values: a. Write expressions using relational operators. b. Evaluate expressions that use relational operators. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements AAP-2.I: For nested selection: a. Write nested conditional statements. b. Determine the result of nested conditional statements. AAP-2.K For iteration: a. Write iteration statements. b. Determine the	

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
	Learners will be able to identify, understand, call, and build functions Learners will create algorithms. Learners will combine and modify existing algorithms. Learners will be able to identify, understand when to use and build code which uses lists	3B-AP-15: Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution. 3A-AP-16: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions. 3A-AP-18: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. 3B-AP-14: Construct solutions to problems using student-created components, such as procedures, modules and/or objects.	result or side effect of iteration statements. AAP-3.B: Explain how the use of procedural abstraction manages complexity in a program. CRD-2.B: Explain how a program or code segment functions. AAP-3.A: For procedure calls: a. Write statements to call procedures. 3.B b. Determine the result or effect of a procedure call AAP-3.C: Develop procedural abstractions to manage complexity in a program by writing procedures.	

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible.	AAP-2.L: Compare multiple algorithms to determine if they yield the same side effect or result AAP-2.M: For algorithms: a. Create algorithms. B. Combine and modify existing algorithms. AAP-1.C: Represent a list or string using a variable. 3.A AAP-1.D For data abstraction: a. Develop data abstraction using lists to store multiple elements 3.B b. Explain how the use of data abstraction manages complexity in program code. 3.C AAP-2.N For list operations: a. Write expressions that use list indexing and	

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
			list procedures. 2.B b. Evaluate expressions that use list indexing and list procedures. 4.B	
Lesson 13: Planning a Product	Learners will design a program and its user interface. Learners will develop a program using a development process.	3B-AP-17: Plan and develop programs for broad audiences using a software life cycle process. 3B-AP-21: Develop and use a series of test cases to verify that a program performs according to its design specifications. 3B-AP-23: Evaluate key qualities of a program through a process such as a code review. 3A-AP-19: Systematically design and develop programs for broad audiences by incorporating feedback from users.	CRD-2.E: Develop a program using a development process CRD-2.F: Design a program and its user interface.	8 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
Lesson 14: Building the Product	Learners will build a game using all the programming knowledge previously learned such as variables, operators, logic, loops, collections, algorithms, and functions.	3B-AP-10: Use and adapt classic algorithms to solve computational problems. 3B-AP-12: Identify and Use Operators in JavaScript 3A-AP-17: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 3A-AP-13: Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	AAP-1.A: Represent a value with a variable AAP-2.E: For relationships between two variables, expressions, or values: a. Write expressions using relational operators. b. Evaluate expressions that use relational operators. AAP-2.H: For selection: a. Write conditional statements b. Determine the result of conditional statements AAP-2.I: For nested selection: a. Write nested conditional statements. b. Determine the result of nested conditional statements. AAP-2.K For iteration: a. Write iteration statements. b. Determine the	15 hrs

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3B-AP-15: Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution. 3A-AP-16: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions. 3A-AP-18: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. 3B-AP-14: Construct solutions to problems using student-created components, such as procedures, modules and/or objects.	result or side effect of iteration statements. AAP-3.B: Explain how the use of procedural abstraction manages complexity in a program. CRD-2.B: Explain how a program or code segment functions. AAP-3.A: For procedure calls: a. Write statements to call procedures. 3.B b. Determine the result or effect of a procedure call AAP-3.C: Develop procedural abstractions to manage complexity in a program by writing procedures.	

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
		3A-AP-21: Evaluate and refine computational artifacts to make them more usable and accessible. 3B-AP-17: Plan and develop programs for broad audiences using a software life cycle process. 3B-AP-21: Develop and use a series of test cases to verify that a program performs according to its design specifications. 3B-AP-23: Evaluate key qualities of a program through a process such as a code review.	AAP-2.L: Compare multiple algorithms to determine if they yield the same side effect or result AAP-2.M: For algorithms: a. Create algorithms. B. Combine and modify existing algorithms. AAP-1.C: Represent a list or string using a variable. 3.A AAP-1.D For data abstraction: a. Develop data abstraction using lists to store multiple elements 3.B b. Explain how the use of data abstraction manages complexity in program code. 3.C AAP-2.N For list operations: a. Write expressions that use list indexing and	

Lesson Title	Lesson Objectives	CSTA Standards	AP CSP Learning Objective	Average Completion Time (in hours)
			list procedures. 2.B b. Evaluate expressions that use list indexing and list procedures. 4.B CRD-2.E: Develop a program using a development process CRD-2.F: Design a program and its user interface.	