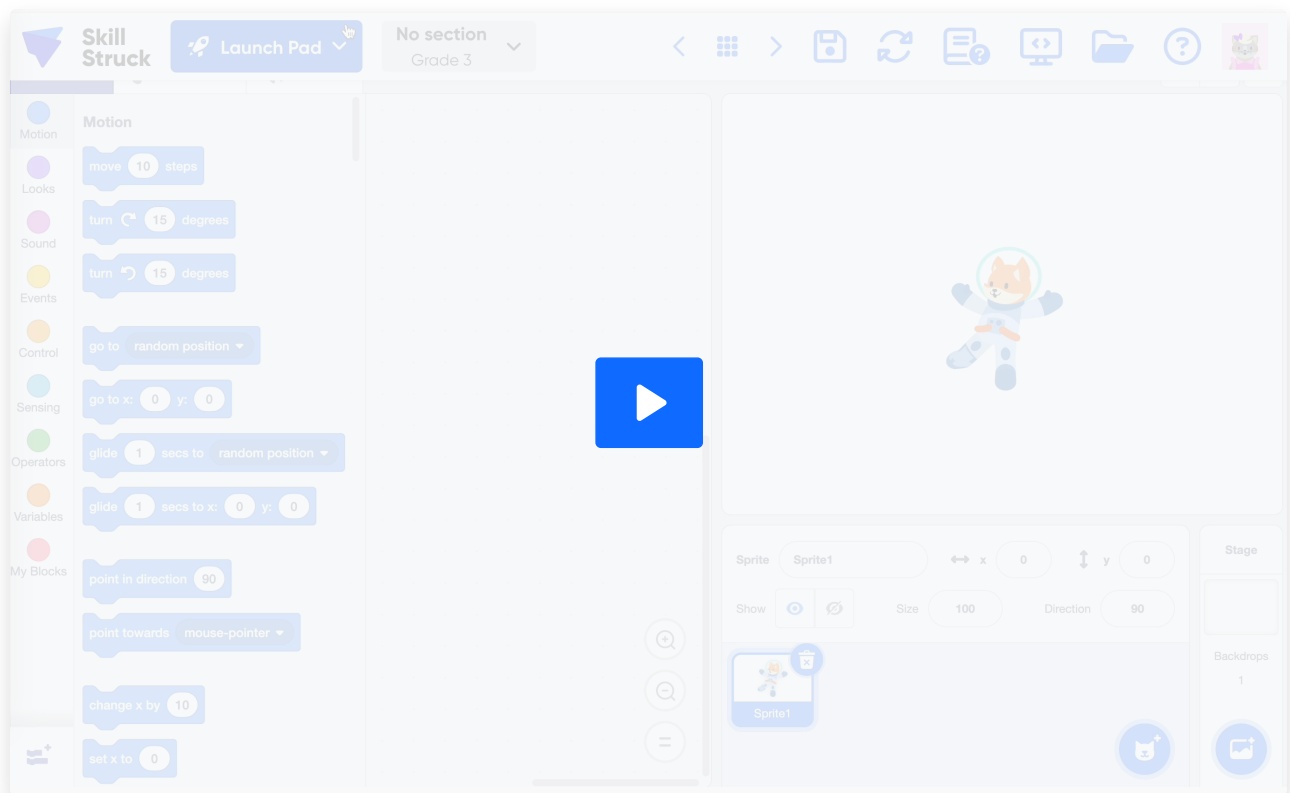


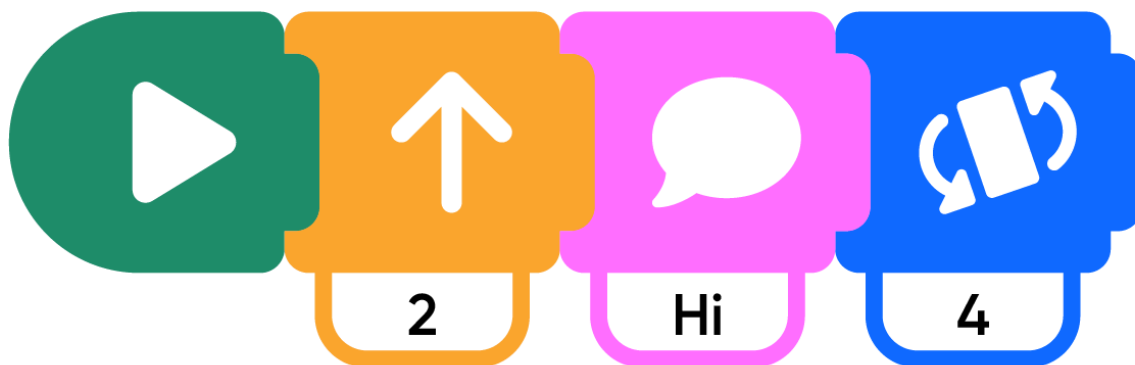
Algorithms and Blocks Pro

Textbook

Algorithms and Blocks Pro

As you read the textbooks remember to read carefully and pay close attention to every word and sentence. Think deeply by asking yourself, "What is this trying to teach me?" or "How does this connect to what I already know?" Finally, be sure you grasp the main points of each section, truly understanding the big ideas!





Every part of your life has algorithms in it! From sports, to art, to cooking, to amusement parks, to getting out of bed, everything is related to algorithms. This is because algorithms are step by step instructions. Instructions are what make the world work.

Algorithms

An [algorithm](#) is a step-by-step list of instructions. Here is an example of an algorithm for making toast.

1. Pick the bread you want
2. Pull out 1 or 2 slices of bread
3. Place the bread in the toaster slots
4. Push down the toaster lever and select your toast preference
5. Wait for the toaster to ding
6. Eat!

Sequencing

What does [sequencing](#) have to do with algorithms? Sequencing is the order that things are done. Algorithms are the instructions. Algorithms must have the correct sequence to get the correct result. Sequencing and algorithms go hand-in-hand. For example, if you tried to "Eat" your toast before you "Pushed down the toaster lever," it wouldn't work! The steps must be in the correct order for *that specific algorithm* to succeed.

Different Paths, Same Destination

Now, here's a fascinating thing about algorithms: while sequencing is super important *within* one set of instructions, there can often be many different sets of instructions (many different algorithms) that all lead to the exact same final result!

Think about two friends trying to get to the same playground from their houses. One friend might take a long route through the park, enjoying the scenery, while the other might take a shorter, more direct path down the street. Both friends end up at the playground, even though they took very different journeys! Programming is often like this.

There are many ways to write code to achieve the exact same outcome. One programmer might use a few long steps, while another might break it down into many smaller steps, or even use different commands entirely. As you complete some of the challenges, practice different commands and see the same outcome!

Putting Things in Order: Sorting!

Have you ever had a pile of messy toys, or a stack of papers mixed up in your backpack? When things are messy, it's hard to find what you're looking for, right? That's where sorting comes in! Sorting means putting information or items into a useful order. This could be alphabetical order (like names in a list), numerical order (like numbers from smallest to largest), or even by color or size.

Sorting is a very important type of algorithm. Just like making toast, sorting requires a step-by-step process. For example, to sort a list of numbers from smallest to largest, you would follow specific instructions to compare numbers and move them into the correct spots until everything is perfectly in order. Like having different paths for the same destination, there are actually many different ways – many different algorithms – that computers can use to sort information, but they all end up with the same helpful result: everything neatly organized!

Sprites

A sprite is a character or object that can move, interact, and have its own code.

To add a sprite:

1. Click the 'Choose Sprite' icon at the bottom right of your screen.
2. Select a sprite from the sprite library.
3. Click on the sprite you want to code and begin.



Backdrops

A backdrop is the background image that your code shows on.

To add a backdrop:

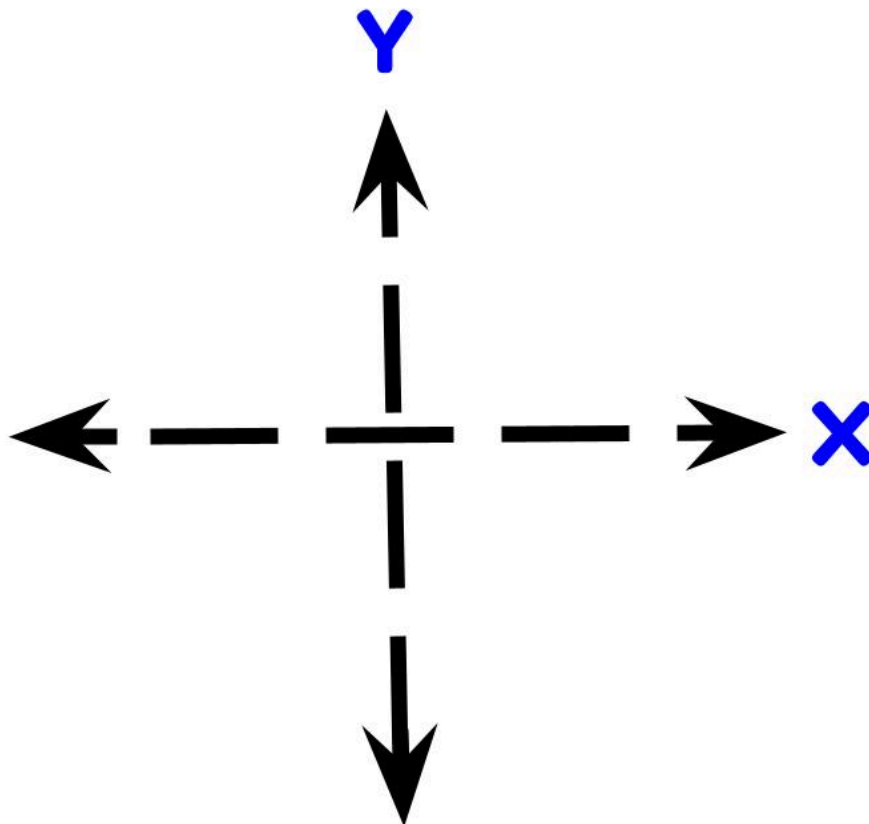
1. Click the 'Choose Backdrop' icon at the bottom right of your screen.
2. Select a backdrop from the backdrop library.



Motion

Motion is when something moves. There are lots of ways to code with motion.

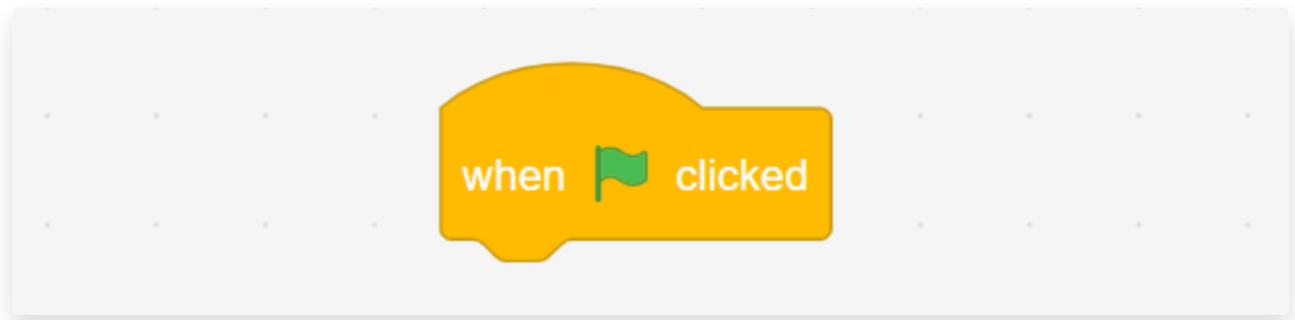
1. Move [10] steps: This block will move your sprite to the right 10 spaces. Change the number to -10 to move left.
2. Change X by [10]: 'X' represents left and right. This block will also move your sprite right 10 spaces. Change the number to -10 to also move left.
3. Change Y by [10]: 'Y' represents up and down. This block will move your sprite up 10 spaces. Change the number to -10 to move down.



Adding an Event

You have to start your code with an event so the computer knows when to start your code.

1. Add an event to your sprite by clicking on the event category.
2. Drag the 'When green flag clicked' sprite and connect it to your motion block. This tells your code to play when you click the green flag.



Critical Thinking Questions

1. Why do you think it's important to follow the steps in the right order when using an algorithm, like getting ready for school or making toast?
2. If you were coding a sprite to move, why would it be important to use the right blocks for motion, like moving left or right, or up and down? What might happen if you used the wrong block?

Questions (5)

1. What is an algorithm?

MULTIPLE CHOICE

Choose the correct answer:

- A. Step by step instructions
- B. The pacing of a program
- C. A computer program that generates music
- D. A form of debugging

2. True or False: algorithms must also have correct sequencing to work right.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

3. True or False: Algorithms only exist in computer programming.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

4. Why is it necessary to add an event to your code?

Choose the correct answer:

- A. To change the sprite's appearance
- B. To add motion to the sprite
- C. To tell the computer when to start running the code
- D. To change the backdrop

5. What is a sprite?

Choose the correct answer:

- A. A step-by-step instruction
- B. A type of backdrop
- C. A character or object that can move and interact
- D. A motion command

Games (3)

1. Algorithms Ordering Game

Full Screen

Audio

Instructions

Answer Key

Pause

Clear All

Check Matches

Attempts: 0

Place the bread in the toaster slots

Wait for the toaster to bing

Eat

Pick the bread you want

Push down the toaster lever and select your toast preference

Pull out 1 or 2 slices of bread



2. Algorithms Typing Game


Full Screen

Audio

Instructions

Restart

Pause



0s 100%

Sequencing is the order t

3. Algorithms Matching Game

Full Screen

Audio

Instructions




Answer Key

Pause

Clear All

Check Matches

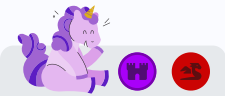
Attempts: 0



Add a background.

When the green flag is clicked event block.

Add a sprite.



Blocks Pro Challenges (4)

1. Walk the Moon

Challenge

Textbook

Walk the Moon

Use the moon background and make it look like your sprite is walking the moon.

Requirements

0/2

1 Motion

1 Event

Blocks must be connected to an Event block in order to pass the requirements

Submit

Code

Costumes

Sounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite

Sprite1

x 0 y 0

Size 100 Direction 90

Sprite1

Stage

Backdrops 1

2. Front Flip

Challenge

Textbook

Front Flip

Program your sprite to make it look like it is doing a front flip.

Hint: You will need to click the green flag multiple times to turn your sprite all the way around.

Requirements

0/2

1 Motion

1 Event

Blocks must be connected to an Event block in order to pass the requirements

Submit

Code

Costumes

Sounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite

Sprite1

x 0 y 0

Size 100 Direction 90

Sprite1

Stage

Backdrops 1

3. Bouncing Off the Walls

Challenge

Textbook

Bouncing Off the Walls

Imagine you just ate a ton of candy and are bouncing off the walls!

Code a person sprite to look like they are bouncing all over the page by using the **go to random position** block.

Hint: You will need to click the green flag for every time you want your sprite to move.

Requirements0/2

1Motion

1Event

Blocks must be connected to an Event block in order to pass the requirements

Submit

CodeCostumesSounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite1

x 0 y 0

Size 100 Direction 90

Sprite1

Stage

Backdrops 1

4. Jump!

Challenge

Textbook

Jump!

Make your sprite look like it is jumping up and down using the **change y by [10]** block to jump up and **change y by [-10]** to jump down.

Note: You will need to use a **wait 1 second** control block between the two **change y by** blocks

Requirements0/3

2Motion

1Event

1Control

Blocks must be connected to an Event block in order to pass the requirements

Submit

CodeCostumesSounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite1

x 0 y 0

Size 100 Direction 90

Sprite1

Stage

Backdrops 1

Answer Keys & Solutions

Questions

1. What is an algorithm?

MULTIPLE CHOICE

Correct Answer:

- | | |
|--|-------------|
| A. Step by step instructions | ✓ Correct |
| B. The pacing of a program | ✗ Incorrect |
| C. A computer program that generates music | ✗ Incorrect |
| D. A form of debugging | ✗ Incorrect |

Explanation:

You must follow these in order.

2. True or False: algorithms must also have correct sequencing to work right.

MULTIPLE CHOICE

Correct Answer:

- | | |
|----------|-------------|
| A. True | ✓ Correct |
| B. False | ✗ Incorrect |

3. True or False: Algorithms only exist in computer programming.

MULTIPLE CHOICE

Correct Answer:

- | | |
|----------|-------------|
| A. True | ✗ Incorrect |
| B. False | ✓ Correct |

Explanation:

Do you only follow steps in computer programming, or do you also follow steps in real life?

4. Why is it necessary to add an event to your code?

Correct Answer:

- A. To change the sprite's appearance ✗ Incorrect
- B. To add motion to the sprite ✗ Incorrect
- C. To tell the computer when to start running the code ✓ Correct
- D. To change the backdrop ✗ Incorrect

Explanation:

Events trigger the execution of the code.

5. What is a sprite?

Correct Answer:

- A. A step-by-step instruction ✗ Incorrect
- B. A type of backdrop ✗ Incorrect
- C. A character or object that can move and interact ✓ Correct
- D. A motion command ✗ Incorrect

Explanation:

Sprites can move and interact with other elements.

Games

1. Algorithms Ordering Game

Correct Order:

1. Pick the bread you want
2. Pull out 1 or 2 slices of bread
3. Place the bread in the toaster slots
4. Push down the toaster lever and select your toast preference
5. Wait for the toaster to bring
6. Eat

Scoring:

- Gold: 1 attempts or fewer
- Silver: 2 attempts or fewer
- Bronze: 3 attempts or fewer

Students must arrange items in the correct sequence.

2. Algorithms Typing Game

Typing game - no answer key needed. Students practice typing the provided content.

3. Algorithms Matching Game

Matching Game Solutions:

1.  A blue circle with a cat icon inside with a plus symbol. →
2.  A blue circle with a picture icon inside with a plus symbol. →
3.  A yellow code block with a green flag that says "when clicked" →

Students must drag items from the left to match with corresponding items on the right.