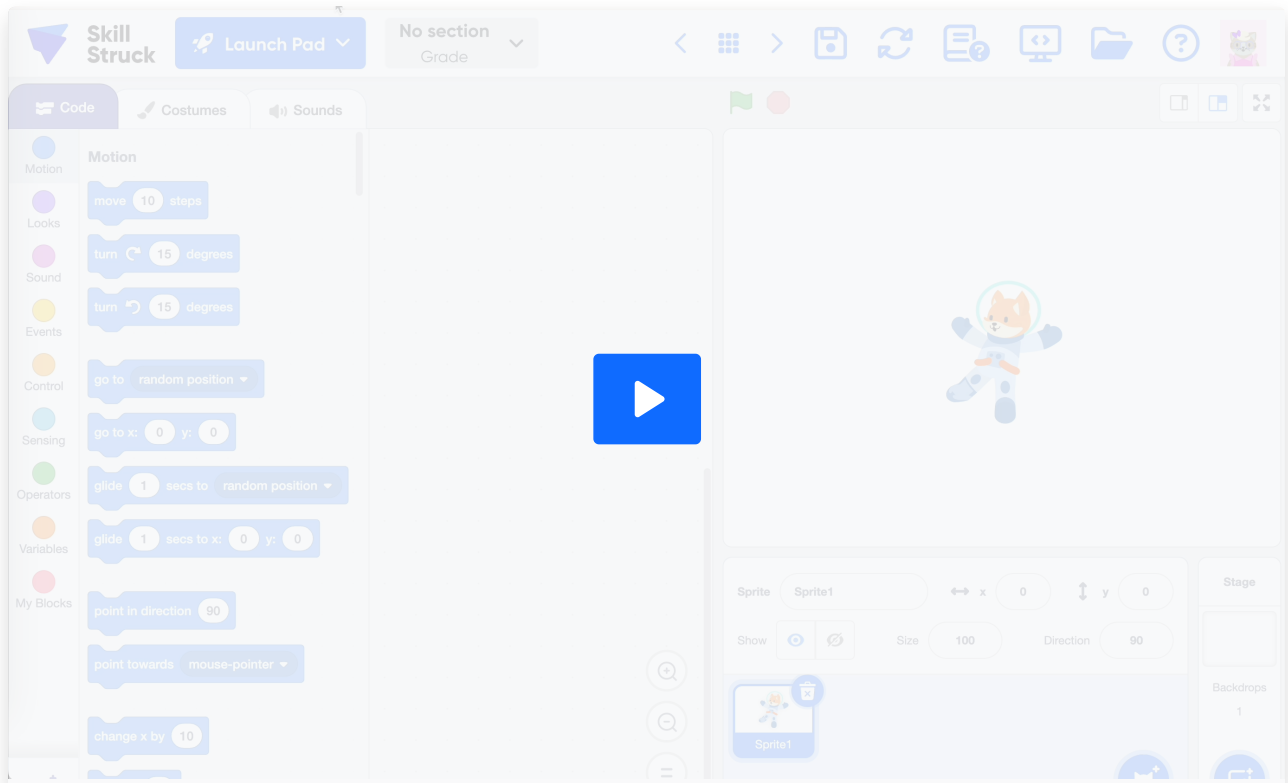
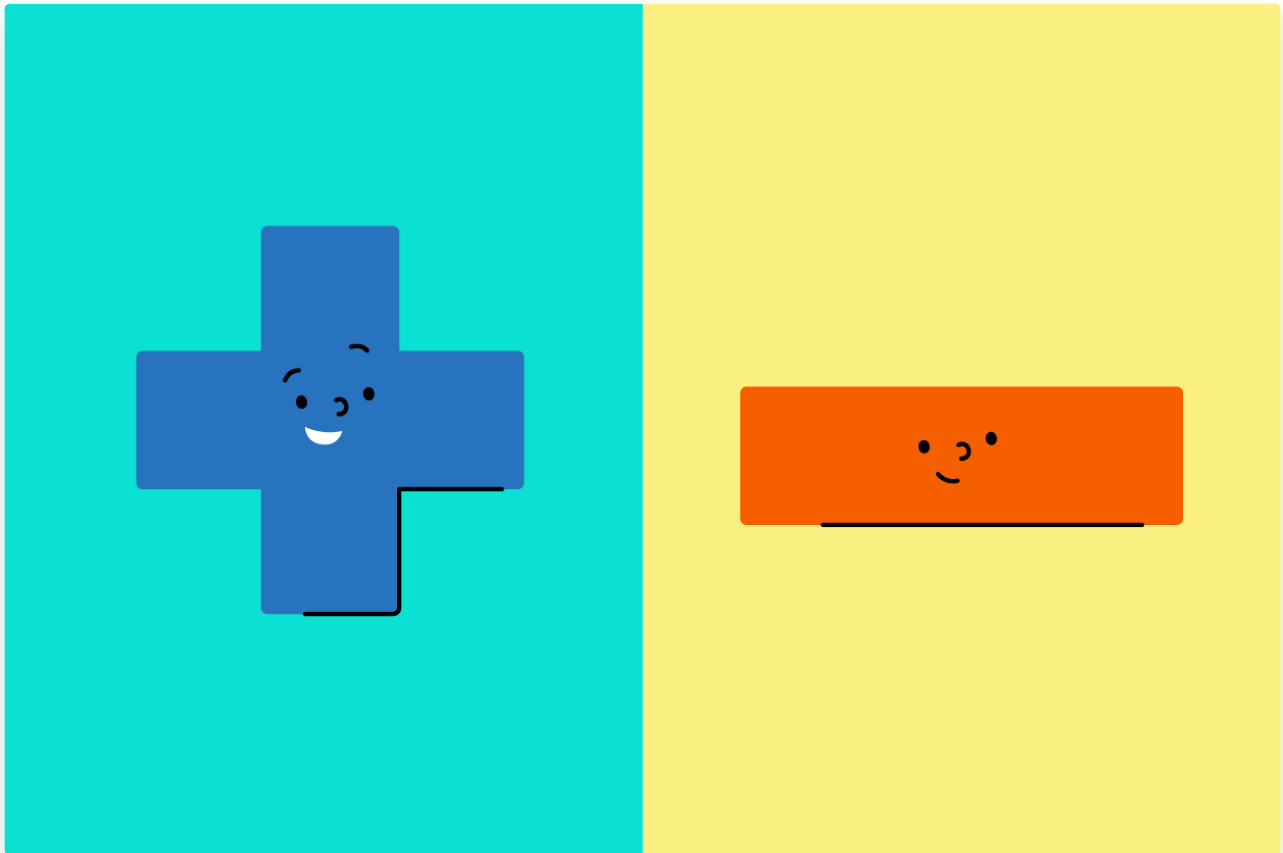


Operators

Textbook

Operators

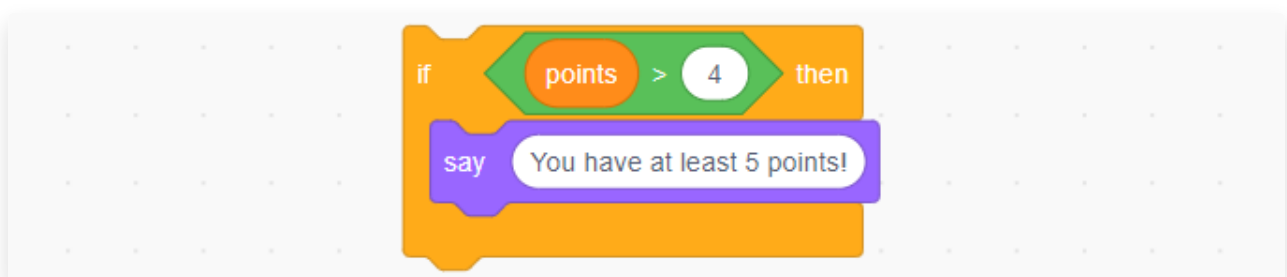




What if you had a program that could do your math homework for you? Computer programs are expert at working with math and calculations. They can add, subtract, multiply, and divide numbers super fast! But even though computers can do the calculations, we still need to understand the math to tell the computer what to do. Let's learn how to program your computer to do math for you!

Operator Blocks

The operator blocks are used to compare [variables](#) and [values](#) and do calculations with numbers.




Thinking Like a Programmer-Mathematician

When you're learning about these operators, remember to **think like a mathematician!**

- **Before you start:** What do you already know about adding or comparing numbers? How might the computer see these problems differently than you do?
- **Don't give up!** If a problem seems tricky, that's okay! Try to break it into smaller parts. What's the first step the computer needs to do?
- **Ask questions:** If you're confused, ask yourself: "What is this operator trying to tell the computer to do?" or "How would I solve this with pencil and paper first?"

Basic Operations

Operations are used to do calculations. Some operations you may be familiar with are addition, subtraction, multiplication, and division. These are sometimes represented with the following symbols:

- **Addition (+):** Joining two groups together.
 - **Example:** If you have 5 apples and get 3 more, how many do you have? You know $5+3=8$.
 - **Represent It:** How would you show $5+3=8$ using drawing or counting blocks? How does the  symbol represent that idea of "joining"?
- **Subtraction (-):** Taking one group away from another.
 - **Example:** If you have 10 cookies and eat 4, how many are left? $10-4=6$.
 - **Fluency Check:** Can you quickly tell me what $12-7$ is without counting on your fingers? Practicing these quick math facts helps you think faster when you're coding too!
- **Multiplication (*):** Combining equal groups. (In coding, we often use an asterisk `*` for multiplication).
 - **Example:** If you have 3 bags with 5 candies in each bag, how many candies total? That's $3*5=15$.
 - **Reasonableness Check:** If you multiply two numbers, and your answer is smaller than *both* of the numbers you started with, does that seem right? Why or why not?
- **Division (/):** Sharing a total number into equal groups. (In coding, we often use a slash `/` for division).
 - **Example:** If you have 12 stickers to share equally among 4 friends, how many stickers does each friend get? $12/4=3$.

Inequalities and the Equal Sign

Operations also include **inequalities**. Inequalities are like a math sentence that uses symbols to compare two things.

- **Greater than (>):** Shows that one number is bigger than another.
 - **Example:** $7>3$ (7 is greater than 3).
 - **Talk About It:** Think about a time you compared two groups of something. How is saying "I have more cookies than you" like using the "greater than" symbol?
- **Less than (<):** Shows that one number is smaller than another.
 - **Example:** $3<7$ (3 is less than 7).
- **Equal to (=):** Shows that two numbers or amounts are exactly the same.
 - **Example:** $5=5$ or $2+3=5$.
 - **Reasonableness Check:** If your code checks if a score is equal to 10, and someone's score is 9, should the program say they won? Why not? How does making sure things are "equal" help

your program make fair decisions?

- **Reflect and Compare:** How is the "equals" sign different from the "greater than" or "less than" signs? When would you choose to use one over the others in a program?

On the coding platform, all these symbols (+, -, *, /, >, <, =) are called **operators**. Understanding them helps you write smart code that can do amazing things with numbers!

Critical Thinking Questions

1. Why do you think it's important for people to be able to do math quickly and without a calculator in everyday life?
2. Imagine you're trying to figure out how many snacks each friend gets if you share 15 snacks among 3 friends. How could you draw a picture to show this problem? What operator would you use in a computer program to solve it?
3. If your computer program needs to check if a player has enough points to get to the next level (say, 50 points), which operator would you use to compare their current points to the 50 points needed? Explain why.

Questions (5)

1. True or False: Operator blocks add letters.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

2. True or False: Operators compare variables.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

3. Which symbol indicates that two numbers are equal to each other in programming?

MULTIPLE CHOICE

Choose the correct answer:

- A. >
- B. <
- C. /
- D. =

4. What operation does the symbol "/" represent in programming?

Choose the correct answer:

- A. Multiplication
- B. Division
- C. Subtraction
- D. Addition

5. What are operator blocks used for in programming?

Choose the correct answer:

- A. To draw shapes on the screen
- B. To compare variables and values and perform calculations
- C. To play music in the background
- D. To change the color of sprites

Games (3)

1. Operators Typing

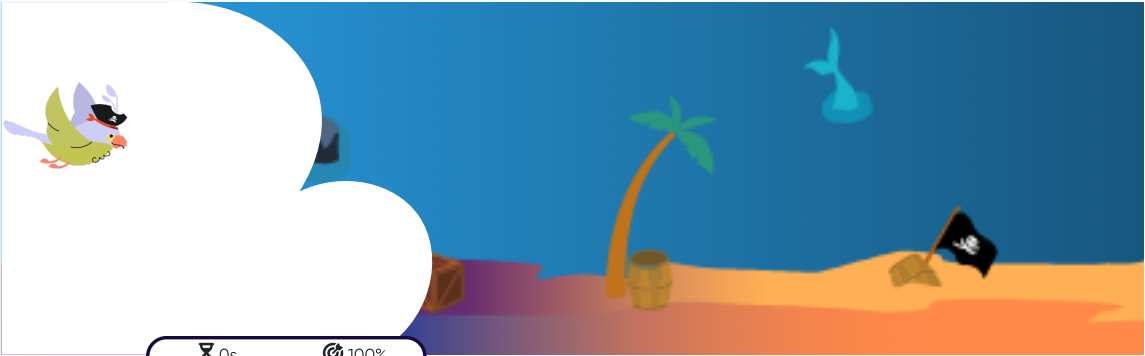
Full Screen

Audio

Instructions

Restart

Pause



0s 100%

Operations are used to d

2. Operators Matching Game

Full Screen

Audio

Instructions

Answer Key

Pause

Clear All

Check Matches

Attempts: 0

Addition

Subtraction

Multiplication

Division

Less Than

Greater Than

Equal To

*

>




<

+

/

-

=



3. Operators Memory Game

Full Screen

Audio

Instructions

Answer Key

Pause

Flips: 0

1 >

2 +

3 -

4 \

5 <

6 +

7 -




8 *

9 <

10 +

11 \

12 >



Blocks Pro Challenges (3)

1. Apple Picking

Apple Picking

Create a game where a person goes apple picking. The goal is to collect 3 apples. To do this:

First, make a variable called "Apple."

Program the collector sprite to respond to all arrow controls. Then, code this sprite so when the green flag is clicked it will say "You did it!" IF the variable "Apple" is **equal to** 3. Also, code this sprite so when the green flag is clicked it will say "Keep going" if the variable "Apple" is **less than** 3. You will need to use the green operator blocks for this.

Program each apple so when the green flag is clicked, the apple will hide and change the variable "apple" by 1 if it is touching your collector sprite.

Note: Your conditional blocks must be in forever blocks for all sprites.

Requirements

0/7

4 Motion

4 Looks

The code editor shows the following blocks in the Motion category:

- 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

The stage shows a cat sprite named Sprite1 with a size of 100 and a direction of 90.

2. 5 Points

5 Points

Create a variable named "Points" and add a sprite to your scene.

Program your code so when the green flag is clicked it will change the variable "Points" by 1.

Then, add a conditional so IF the variable "Points" is greater than ($>$) 4, your sprite will say "You have at least 5 points!"

Note: The greater than ($>$) block can be found in the green operators category.

Requirements

0/5

1 Looks

1 Event

1 Control

1 Operator

2 Variable

The code editor shows the following blocks in the Motion category:

- 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

The stage shows a cat sprite named Sprite1 with a size of 100 and a direction of 90.

3. Look Out for Lightning

Look Out for Lightning

Select a lightning bolt sprite and place it on the scene.

Then, select a new sprite and program it to respond to all arrow controls - up, down, right, left.

Then program it so when the green flag is clicked, the sprite will go to XY coordinate (0, 0) if it is touching the edge or the lightning bolt sprite.

Use the 'or' operator block for this and make sure your conditional is in a forever block.

Requirements

0/5

5 Motion

5 Event

2 Control

2 Sensing

1 Operator

The image shows the Scratch code editor interface. The 'Code' tab is selected, displaying a script for a sprite. The script is as follows:

- When green flag clicked** (Event block)
 - go to random position** (Motion block)
 - go to x: 0 y: 0** (Motion block)
 - glide 1 secs to random position** (Motion block)
 - glide 1 secs to x: 0 y: 0** (Motion block)
 - point in direction 90** (Motion block)
 - point towards mouse-pointer** (Motion block)
 - change x by 10** (Motion block)
 - set x to 0** (Motion block)
 - change y by 10** (Motion block)
 - set y to 0** (Motion block)
 - if on edge, bounce** (Control block)
 - set rotation style left-right** (Motion block)
- when arrow key pressed** (Event block)
 - if touching lightning bolt sprite or touching edge** (Sensing block)
 - go to random position** (Motion block)
 - go to x: 0 y: 0** (Motion block)
 - glide 1 secs to random position** (Motion block)
 - glide 1 secs to x: 0 y: 0** (Motion block)
 - point in direction 90** (Motion block)
 - point towards mouse-pointer** (Motion block)
 - change x by 10** (Motion block)
 - set x to 0** (Motion block)
 - change y by 10** (Motion block)
 - set y to 0** (Motion block)
 - if on edge, bounce** (Control block)
 - set rotation style left-right** (Motion block)

The right side of the interface shows the 'Sprite' panel with 'Sprite1' selected. The 'Stage' panel shows the 'Backdrops' list with '1' selected. The 'x position' checkbox is checked.

Answer Keys & Solutions

Questions

1. True or False: Operator blocks add letters.

MULTIPLE CHOICE

Correct Answer:

A. True

✗ Incorrect

B. False

✓ Correct

Explanation:

Operators do calculations with numbers.

2. True or False: Operators compare variables.

MULTIPLE CHOICE

Correct Answer:

A. True

✓ Correct

B. False

✗ Incorrect

Explanation:

An operator can be programmed so if the variable 'score' is less than 5, it says, "The score is less than 5!"

3. Which symbol indicates that two numbers are equal to each other in programming?

MULTIPLE CHOICE

Correct Answer:

A. >

✗ Incorrect

B. <

✗ Incorrect

C. /

✗ Incorrect

D. =

✓ Correct

Explanation:

Consider the symbols to show equal

4. What operation does the symbol "/" represent in programming?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-------------------|-------------|
| A. Multiplication | ✗ Incorrect |
| B. Division | ✓ Correct |
| C. Subtraction | ✗ Incorrect |
| D. Addition | ✗ Incorrect |

Explanation:

Recall the list of basic operations and their symbols.

5. What are operator blocks used for in programming?

MULTIPLE CHOICE

Correct Answer:

- | | |
|---|-------------|
| A. To draw shapes on the screen | ✗ Incorrect |
| B. To compare variables and values and perform calculations | ✓ Correct |
| C. To play music in the background | ✗ Incorrect |
| D. To change the color of sprites | ✗ Incorrect |

Explanation:

Consider the purpose of operator blocks described in the introduction.

Games**1. Operators Typing**

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

2. Operators Matching Game

Matching Game Solutions:

1. \rightarrow
2. \rightarrow
3. \rightarrow
4. \rightarrow
5. \rightarrow
6. \rightarrow
7. \rightarrow

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

3. Operators Memory Game

Memory Game Pairs:

1. \leftrightarrow
2. \leftrightarrow
3. \leftrightarrow
4. \leftrightarrow
5. \leftrightarrow
6. \leftrightarrow

Students must find all matching pairs by flipping cards and remembering their positions.