

Algorithms in Computer Programming

Textbook

Algorithms in Computer Programming

Algorithms are important in computer programming. An algorithm tells the computer what to do and in what order so that the computer will do what you want it to do. When you press the "play" button on a video, you want the computer to start playing the video. Luckily, there is an algorithm in place that tells the computer that when you click the "play button" it will follow instructions that will make the video start.

ALGORITHMS

Code Castle



Skill Struck™

▶ 0:00 / 3:00



Practice Building Algorithms!



In the puzzle game, you will have different options to move your character. These are different JavaScript commands. JavaScript is a programming language.

Options to Move Your Character

```
moveForward();  
turnRight();  
turnLeft();
```

Choose from these commands to direct your character to the puzzle's finish line. Make sure to put the steps into the correct sequence to build an algorithm that will work. You can use commands more than once.

```
1 moveForward();  
2 turnRight();  
3 turnLeft();
```

These statements need to be put in the correct order or sequence. If the code statements are placed out of order, the character will not make it to where it needs to go.

Remember that these movements are **from the character's point of view!** So turnRight means that according to the character, he will turn right. This will take some thinking to make sure the sequence is in the right order.

Each of these commands are called *code statements*. Sequencing is the application of each step of an algorithm in the order in which the code statements are given. The sequence or order of these commands matters.

A Solution to the Puzzle Above

```
turnLeft();  
moveForward();  
moveForward();
```

Predict

Before running your code, make a prediction of what your code is going to do.

Comparing Methods to Show Algorithms

As you practice moving your knight through the puzzle, you are building an algorithm! There are many ways to represent the same algorithm. We'll explore a few ways to show the same algorithm.

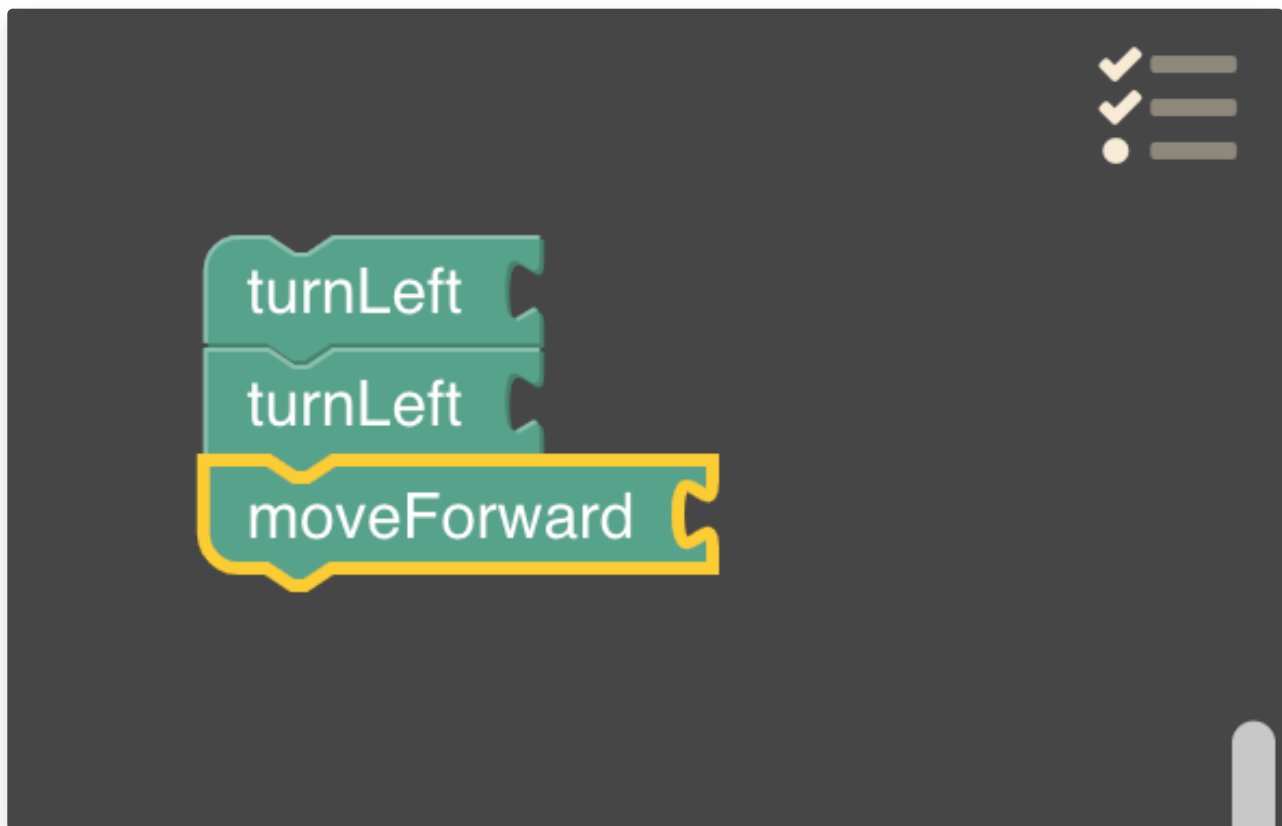
Let's take the algorithm above. One way to represent this algorithm is to use code, like in the image above.

```
turnLeft();
```

```
moveForward();
```

```
moveForward();
```

Another way is to use blocks.



You can toggle back and forth between the two methods by clicking on the checklist icon in the top right corner.

Another method is to just describe what the algorithm is doing.

Turn left twice and then move forward.

Each of these methods is an effective way to represent an algorithm. Different methods might be better suited for different situations.

Readability

Remember to keep your code environment clean! Only have the blocks or code segments you are actually using on your screen. Clarity and readability are important considerations when expressing an algorithm in a programming language.

Connecting Patterns

Algorithms aren't just for computers; you use them every day! Think about following a recipe: you need to add ingredients in a specific order, or your dish won't turn out right. Or consider giving someone directions: "Turn left at the light, then go straight for two blocks, then turn right." If you mix up those steps, they'll end up somewhere completely different. These real-world examples show how sequences of instructions, or algorithms, are essential for achieving a desired outcome, whether it's cooking a meal, navigating a city, or getting a computer to play a video.

Solving Patterns

Remember that math and programming are all about patterns and structure! Don't just see a problem; look for the relevant details that truly matter. Think like a detective, and try to create plans and procedures that logically order your steps and ideas, breaking down even the most complex problems into manageable parts.

You'll often find that a new concept isn't entirely new; it might relate directly to something you've learned before. By looking for similarities between different problems, you'll build a powerful toolkit that can help you connect the solutions of small problems to more complicated, large-scale situations in the real world.

Does this Make Sense?

Expressing solutions as a series of **computational steps** is fundamental to programming. Instead of trying to solve a problem all at once, effective programmers break it down into smaller, manageable pieces, addressing each part step-by-step. This approach not only makes complex problems easier to tackle but also allows you to represent your solution in various ways—whether through written descriptions, diagrams, or actual code—depending on what best communicates your plan. As you develop these step-by-step solutions, you'll start to recognize patterns and structures, helping you connect different programming ideas and apply what you learn to new challenges. Always remember to continually check your work as you go, asking yourself, "Does this make sense?" and justifying why each step contributes to the overall correct solution.

Questions (6)

1. What is the term that refers to putting the steps into the correct order?

MULTIPLE CHOICE

Choose the correct answer:

- A. sequencing
- B. conditional
- C. progression
- D. cascade

2. Which point of view are the puzzle algorithms running from?

MULTIPLE CHOICE

Choose the correct answer:

- A. The programmer's point of view
- B. The character's point of view

3. Each of these lines of code are called what?

MULTIPLE CHOICE

`moveForward(); turnRight(); turnLeft();`

Choose the correct answer:

- A. code commands
- B. puzzle statement
- C. algorithm command
- D. sequence step

4. What are some effective ways to represent algorithms? Choose 2.

SELECT MULTIPLE

Select all that apply:

- A. code
- B. blocks
- C. arrows
- D. musical notes

5. True or False: You can only use the same command once in an algorithm.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

6. True or False: Clarity and readability are important considerations when expressing an algorithm in a programming language.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

Puzzles (5)

1. AP Algorithms 1

Instructions

Answer Key

</>

Start

moveForward

turnRight

turnLeft

turnAround

repeat 10 times

do

repeat while

do

if

do

and

not

isClearAhead

isClearLeft

2. AP Algorithms 2

Instructions

Answer Key

</>

Start

moveForward

turnRight

turnLeft

turnAround

repeat 10 times

do

repeat while

do

if

do

and

not

isClearAhead

isClearLeft

3. AP Algorithms 3

Instructions

Answer Key

</>

Start

Start

moveForward

turnRight

turnLeft

turnAround

repeat 10 times

do

repeat while

do

if

do

and

not

isClearAhead

isClearLeft

4. AP Algorithms 4

Instructions

Answer Key

</>

Start

Start

moveForward

turnRight

turnLeft

turnAround

repeat 10 times

do

repeat while

do

if

do

and

not

isClearAhead

isClearLeft

5. AP Algorithms 5

Instructions

Answer Key

`</>`

Start

moveForward

turnRight

turnLeft

turnAround

repeat 10 times
do

repeat while
do

if
do

and

not

isClearAhead

isClearLeft

Start

Answer Keys & Solutions

Questions

1. What is the term that refers to putting the steps into the correct order?

MULTIPLE CHOICE

Correct Answer:

- | | |
|----------------|-------------|
| A. sequencing | ✓ Correct |
| B. conditional | ✗ Incorrect |
| C. progression | ✗ Incorrect |
| D. cascade | ✗ Incorrect |

Explanation:

Putting things in the right order is called sequencing

2. Which point of view are the puzzle algorithms running from?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-----------------------------------|-------------|
| A. The programmer's point of view | ✗ Incorrect |
| B. The character's point of view | ✓ Correct |

Explanation:

TurnRight will turn according to the character on the screen.

3. Each of these lines of code are called what?

MULTIPLE CHOICE

Correct Answer:

- | | |
|----------------------|-------------|
| A. code commands | ✓ Correct |
| B. puzzle statement | ✗ Incorrect |
| C. algorithm command | ✗ Incorrect |

D. sequence step

✗ Incorrect

Explanation:

They are called a command.

4. What are some effective ways to represent algorithms? Choose 2.

SELECT MULTIPLE

Correct Answers:

A. code

✓ Correct

B. blocks

✓ Correct

C. arrows

✗ Incorrect

D. musical notes

✗ Incorrect

Explanation:

You can toggle back and forth between these two methods on the platform.

5. True or False: You can only use the same command once in an algorithm.

MULTIPLE CHOICE

Correct Answer:

A. True

✗ Incorrect

B. False

✓ Correct

Explanation:

The same command can be used as many times as you want.

6. True or False: Clarity and readability are important considerations when expressing an algorithm in a programming language.

MULTIPLE CHOICE

Correct Answer:

A. True

✓ Correct

B. False

✗ Incorrect

Explanation:

Keeping things clean and readable are important to build effective algorithms.

Puzzles

1. AP Algorithms 1

Solution:

```
1 moveForward();
2 turnLeft();
3 moveForward();
4 moveForward();
5 turnRight();
6 moveForward();
7 moveForward();
```

2. AP Algorithms 2

Solution:

```
1 turnLeft();
2 moveForward();
3 moveForward();
4 turnLeft();
5 moveForward();
6 moveForward();
7 turnLeft();
8 moveForward();
9 turnRight();
10 moveForward();
11 moveForward();
12 turnRight();
13 moveForward();
14 moveForward();
```

3. AP Algorithms 3

Solution:

```
1 turnLeft();
2 moveForward();
3 moveForward();
4 turnRight();
5 moveForward();
```

```
6 moveForward();
7 turnRight();
8 moveForward();
9 moveForward();
10 turnLeft();
11 moveForward();
12 moveForward();
13 turnLeft();
14 moveForward();
```

4. AP Algorithms 4

Solution:

```
1 turnRight();
2 moveForward();
3 turnRight();
4 moveForward();
5 moveForward();
6 moveForward();
7 moveForward();
8 turnLeft();
9 moveForward();
10 moveForward();
```

5. AP Algorithms 5

Solution:

```
1 turnRight();
2 moveForward();
3 turnRight();
4 moveForward();
5 turnLeft();
6 moveForward();
7 turnRight();
8 moveForward();
9 turnLeft();
10 moveForward();
11 moveForward();
12 turnLeft();
13 moveForward();
```