

Activity

## Unplugged Coding: Hopscotch

**Level:** Kindergarten–Grade 3 • **Topic:** Computer Fundamentals

*If you are looking for an accessible online version of this content, please visit the [Pinnguaq website](https://pinnguaq.com/learn/unplugged-coding-hopscotch) (link: <https://pinnguaq.com/learn/unplugged-coding-hopscotch>).*

---

### About the Author

#### Kayla Blacquiere

Kayla is currently one of the educators in the Lindsay Makerspace who helps plan, design and deliver curriculum for a variety of programs. She has a long history of working with children in many different areas such as gymnastics, cheerleading, summer camps and public schools. She graduated from The University of Western Ontario with a BA and specialization in Nutrition and Families and also recently completed a BEd at Ontario Tech University in the Primary/Junior division. Her recent studies have led her to discover a love and interest for LEGO robotics, coding and STEAM education.



---

### Overview

This outdoor activity explores the coding concept of **algorithms** through a game of hopscotch. In this, you will create a hopscotch court outside on your driveway or sidewalk and by yourself or with a partner, group of friends or family members, you will create an algorithm to guide them across the hopscotch court. This algorithm will be written inside your ‘code’ box at the top of the court using arrows to direct them to which square they will hop on to. Each person will read your instructions and try their best to follow them to cross the hopscotch board. If they cannot follow the instructions and take a wrong step they must start again!

An **algorithm** is a list or series of steps to complete a specific task. Did you know that we use algorithms everyday without even noticing? For example, each day you brush your teeth and follow the same steps:

1. Go to the sink
2. Get out your toothbrush
3. Get out the toothpaste
4. Turn on the tap (usually cold water)
5. Rinse your toothbrush
6. Put toothpaste on the toothbrush (approximately the size of your pinky fingernail)
7. Brush in circular motions EVERYWHERE for about 2 minutes
8. Rinse your mouth with tap water
9. Rinse your toothbrush
10. Turn off the tap
11. Put your toothbrush and toothpaste back in their designated spots

That is eleven steps that you follow each day, twice a day, without even realizing that this is an **algorithm**! This is not the only algorithm that you follow each day. What other things do you do everyday that could be considered an algorithm? What about making breakfast? Do you pour your cereal or milk first? Should you open the drawer to get a spoon before or after getting a bowl from the cupboard? What about taking your dog for a walk? Typically, you need to put on your shoes, tie your shoes, get the dog leash, attach the leash to your dog, turn the doorknob in order to open the door and walk outside. The order of steps in an algorithm are important in problem solving because the steps are sequential and if not done in proper order the problem will not be solved. Look at the examples above, can these be done in a different order? Can you brush your teeth without first putting toothpaste onto your toothbrush?

During this activity you will get the chance to create your own algorithms that will guide players across the hopscotch court. This activity provides you with the opportunity to explore and create algorithms without the use of technology!

---

## Background Information

Unplugged coding opportunities allow learners to engage in many areas of computational thinking while helping to develop knowledge on the basics of coding concepts without actually using technology. Algorithmic thinking is one method of computational thinking that allows learners to develop a solution to a given problem through sequential rules such as creating a series of steps to get a player across a hopscotch court.

As mentioned by BBC, computational thinking can be broken down into four main concepts: decomposition, pattern recognition, abstraction, and algorithms.

- **Decomposition** is a thought process that allows you to break down a problem in smaller steps.
- **Pattern recognition** is a thought process that allows you to connect the given problem to problems you have previously encountered.
- **Abstraction** is a thought process that allows you to focus on relevant information and problem solve by focusing on your core knowledge surrounding the problem.
- **Algorithmic thinking** is a thought process that allows you to develop a solution to a given problem through sequential rules.

For more information, read the **Introduction to computational thinking article and lesson** (link: <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>) provided by BBC Bitesize.

## Vocabulary

- **Algorithm** - a list or series of steps to complete a specific task.
- **Code** - specific instructions that are given to a computer to tell it what to do.

---

## Materials

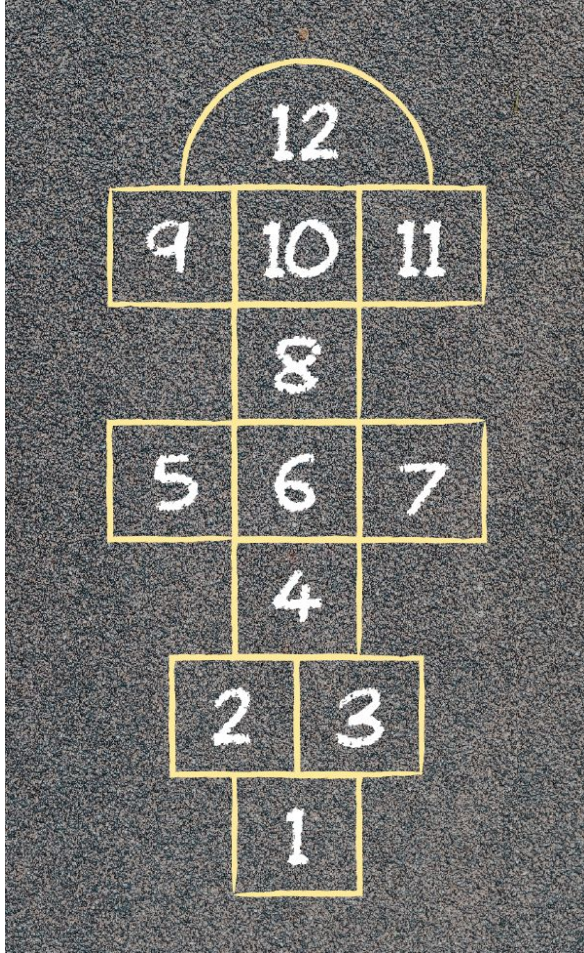
- Sidewalk chalk
- Large outdoor area on pavement or sidewalk

---

## Step-by-Step Instructions

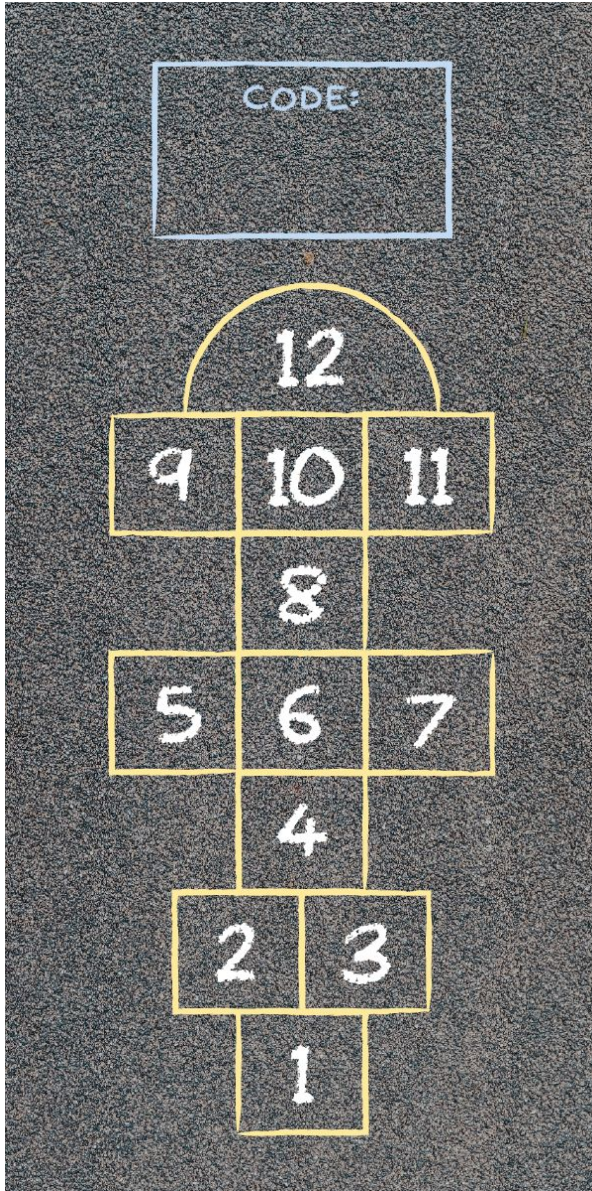
The following step-by-step instructions will guide you through creating, drawing and playing your hopscotch game! Be sure to read all the instructions before you begin.

**Step 1:** Draw your hopscotch court on pavement or on a sidewalk using your sidewalk chalk. Use the following image to see how the hopscotch court should look. The numbers will start at 1 and go up to 12!

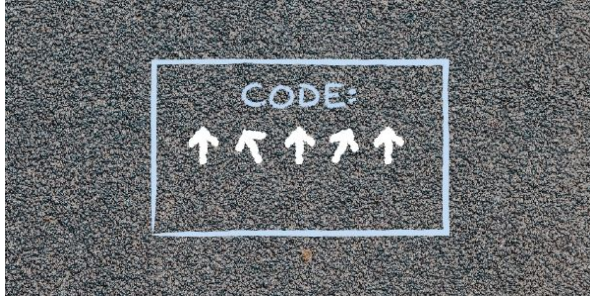


**Step 2:** Draw your code box at the top of the court so that it is easy to see and read. This code box will communicate the algorithm using arrows.





**Step 3:** Create an algorithm that will guide you or your partner across the hopscotch court. Communicate the algorithm using arrows in the code box at the top of the hopscotch court. The arrows will point in the direction that you want your player to hop. Each arrow will represent a single hop or move. Every algorithm will begin with an arrow pointing up as you want the first hop to get your player onto the hopscotch court and into box number one.



**Step 4:** Follow the algorithm inside the code box. If you are directed to jump onto a box in a row on its own, use only one foot. If you are directed to jump onto a box that is in a row of two or three boxes, land on both feet inside the box you must go to. If you go to the wrong box or use the incorrect number of feet inside a box, start over!

**Step 5:** How many different algorithms can you or your friends and family create to direct you across the hopscotch court? Take turns writing algorithms and following the algorithms inside the code box!

---

## Follow Up

We want to see the awesome things you're creating! Share your work with us online by tagging @pinnguaq on [Facebook](#), [Twitter](#) or [Instagram](#). Don't forget to include the hashtag #LearnWithPinnguaq! You can also send us your work by email at [media@pinnguaq.com](mailto:media@pinnguaq.com).

For more fun activities, visit [pinnguaq.com/learn](http://pinnguaq.com/learn).

---

## Resources

- BBC Bitesize: [Introduction to computational thinking article and lesson](https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1) (link: <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>)