

## Puzzle Pack 2



for kids, with kids, by kids.

## About this guide

This guide is for teachers and club volunteers and provides a solution to completing the puzzles. For this reason, **this document should not be shared with children**. This document also outlines the learning objectives of each puzzle.



## Introduction

In this project, children will learn to use parameters in functions and manipulate JavaScript values. They will also learn how to build more complex game puzzles. To give the activities some context, the children are building puzzles to protect their land from invasion by Glitches. Each puzzle should be solvable so that friendly animals can move around the player's kingdom.

## Resources

This project requires the use of Code Kingdoms, which is best used in Google Chrome. You can find the website at [codekingdoms.com/codeclub](https://codekingdoms.com/codeclub)

## Learning Objectives

### JavaScript

Constant parameters in functions: I understand how parameters can affect the output of a function.

Changing values (numbers, strings, Booleans): I understand the difference among different types of values and can explain in what situations I might use each.

### Computational Thinking

A3: Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment).

E2: Assessing whether an algorithm does the right thing (functional correctness).

### Computing Progression

YA1: I know that algorithms are implemented on digital devices as programs.

YP2: I can use logical reasoning to predict the behaviour of programs.

PA2: I know that computers need precise instructions.

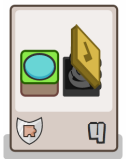
PP3: I can run, check and change programs.

## Challenges



Back on Track

Manipulate values for BoostPads to create a maze puzzle



Over the Hill

Code a catapult puzzle to fling characters across the screen



Pathfinder

Code a more complex maze puzzle with multiple BoostPads



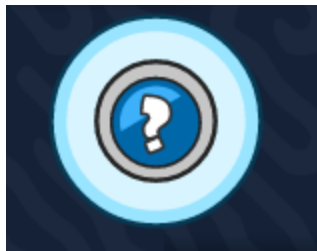
Glitch-a-pult

Code multiple catapults to fling Glitches around the screen

## Frequently Asked Questions

Q. What other help is there for the player?

- a. To help visualise what a puzzle should look like, players can click on the Question Mark icon to see a completed version.



- b. If a player is unsure which steps of the puzzle they have completed or what still needs doing, they can click on the checklist icon to see a list of the required steps.



Q. What is the purpose of building puzzles?

- a. Building puzzles is a defence against invading Glitches. The puzzles, however, must be solvable by friendly animals so they can move freely around your kingdom. Glitches aren't very clever so if a puzzle is coded well they are unlikely to be able to solve it. The general rule is if an animal tester can bypass a puzzle piece (e.g. catapult) to reach the checkpoint then a Glitch will easily get past the puzzle.

# Back on Track

Constant parameters in functions

Changing values

A3

E2

YA1

YP2

PA2

PP3



## Description

Back On Track introduces the player to BoostPads and the ability to manipulate values, in this case directional values. These BoostPads have a single arrow and will propel characters one square.

## Design Tip

Use the water block to turn the puzzle area into a big lake before starting to design the path across it. This will make it easier to plan a route that works with the BoostPads.

## Steps to complete



Build a lake with a narrow path across it.



Place the BoostPad along the path. In this state it should direct the Glitch stepping on the BoostPad into the water.



Place the Blue Button in the puzzle area.

4

Code the obstacle to be solved when the button is pressed. Pressing the button should also trigger changing the direction of the BoostPad so the tester can proceed.

5

Test the puzzle using the play icon in the bottom right corner of the screen.

## Code required to complete the puzzle

To correctly build this puzzle you will need to write the code shown on the right. The BoostPad should change direction when the button is pressed to allow the tester to reach the checkpoint at the end of the puzzle area.

```
ButtonA  
function onPress() {  
  obstacle.solve();  
  BoostPadA.direction = NORTH ;  
}
```



## Skill Development

### JavaScript

Constant parameters in functions: I understand how parameters can affect the output of a function.

Changing values (numbers, strings, Booleans): I understand the difference among different types of values and can explain in what situations I might use each.

### Computational Thinking

A3: Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment).

E2: Assessing whether an algorithm does the right thing (functional correctness).

### Computing Progression

YA1: I know that algorithms are implemented on digital devices as programs.

YP2: I can use logical reasoning to predict the behaviour of programs.

PA2: I know that computers need precise instructions.

PP3: I can run, check and change programs.



# Over the Hill

Constant parameters in functions

Changing values

A3

E2

YA1

YP2

PA2

PP3



## Description

Over the Hill introduces the player to catapults and also reinforces manipulating values learned in Back on Track.

## Design Tip

These catapults will fling characters three squares. This value can't be changed for this puzzle.

## Steps to complete



Build a hill to act as a barrier.



Place the catapult near the hill. In this state it should direct the Glitch stepping on it away from the hill.



Place the Blue Button in the puzzle area.

4

Code the obstacle to be solved when the button is pressed. Pressing the button should also trigger changing the direction of the catapult so the tester is flung over the hill.

5

Test the puzzle using the play icon in the bottom right corner of the screen.

## Code required to complete the puzzle

To correctly build this puzzle you will need to write the code shown below. The catapult should change direction when the button is pressed to fling the tester over the hill and reach the checkpoint at the end of the puzzle area.

```
ButtonA  
function onPress() {  
  obstacle.solve();  
  CatapultA.direction = WEST ;  
}
```



## Skill Development

### JavaScript

Constant parameters in functions: I understand how parameters can affect the output of a function.

Changing values (numbers, strings, Booleans): I understand the difference among different types of values and can explain in what situations I might use each.

### Computational Thinking

A3: Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment).

E2: Assessing whether an algorithm does the right thing (functional correctness).

### Computing Progression

YA1: I know that algorithms are implemented on digital devices as programs.

YP2: I can use logical reasoning to predict the behaviour of programs.

PA2: I know that computers need precise instructions.

PP3: I can run, check and change programs.

# Pathfinder

Constant parameters in functions

Changing values

A3

E2

YA1

YP2

PA2

PP3



## Description

Pathfinder adds a layer of complexity to the BoostPad puzzle created in Back on Track. The double BoostPads will propel characters until they collide with another object so will require more careful design.

## Design Tip

Use the water block to turn the puzzle area into a big lake before starting to design the path across it. This will make it easier to plan a route that works with the BoostPads.

## Steps to complete



Build a lake with a narrow path across it.



Place the BoostPads along the path. In this state they should direct the Glitch stepping on them into the water.

3

Place the Blue Button in the puzzle area.

4

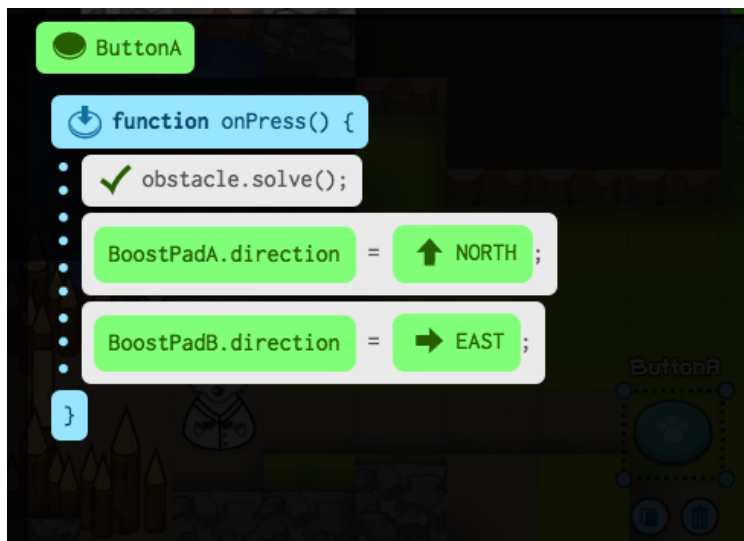
Code the obstacle to be solved when the button is pressed. Pressing the button should also trigger changing the direction of the BoostPads so the tester can proceed.

5

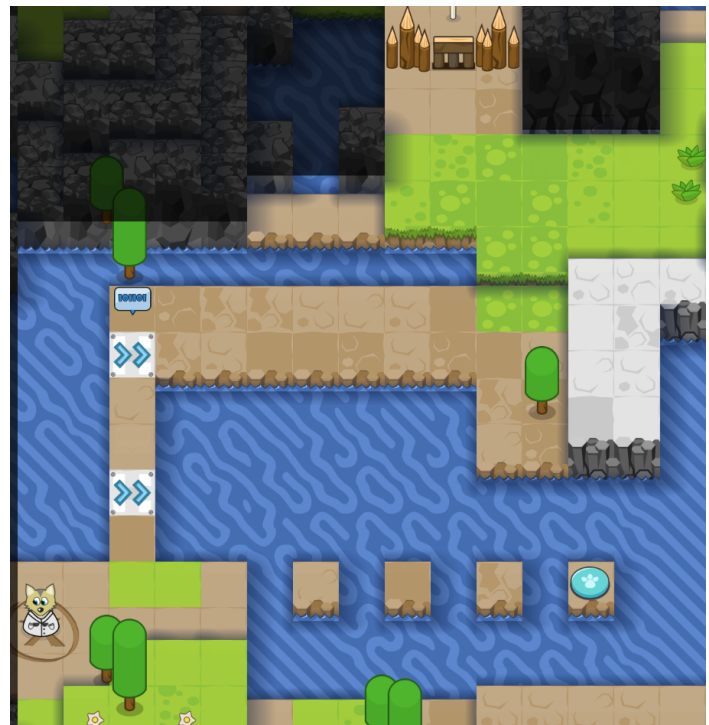
Test the puzzle using the play icon in the bottom right corner of the screen.

## Code required to complete the puzzle

To correctly build this puzzle you will need to write the code shown on below. The BoostPads should change direction when the button is pressed to allow the tester to reach the checkpoint at the end of the puzzle area.



```
ButtonA  
function onPress() {  
  obstacle.solve();  
  BoostPadA.direction = ↑ NORTH ;  
  BoostPadB.direction = → EAST ;  
}
```



## Skill Development

### JavaScript

Constant parameters in functions: I understand how parameters can affect the output of a function.

Changing values (numbers, strings, Booleans): I understand the difference among different types of values and can explain in what situations I might use each.

### Computational Thinking

A3: Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment).

E2: Assessing whether an algorithm does the right thing (functional correctness).

### Computing Progression

YA1: I know that algorithms are implemented on digital devices as programs.

YP2: I can use logical reasoning to predict the behaviour of programs.

PA2: I know that computers need precise instructions.

PP3: I can run, check and change programs.

# Glitch-apult

Constant parameters in functions

Changing values

A3

E2

YA1

YP2

PA2

PP3



## Description

Glitch-apult is similar to Over the Hill but requires a greater level of puzzle design. Players should use three catapults to successfully fling the tester around the puzzle area so they can reach the checkpoint.

## Design Tip

The catapults in this puzzle have distance values that can be manipulated giving the player greater freedom when designing the puzzle.

## Steps to complete



Build three islands, one for each catapult.



Place a catapult on each island.



Place the Blue Button in the puzzle area.

4

Code the obstacle to be solved when the button is pressed. Pressing the button should also trigger changing the direction of the catapult so the tester is flung to the next island.

5

Test the puzzle using the play icon in the bottom right corner of the screen.

## Code required to complete the puzzle

To correctly build this puzzle you will need to write the code shown below. The catapult should change direction when the button is pressed to fling the tester over the hill and reach the checkpoint at the end of the puzzle area.

```

ButtonA
function onPress() {
  obstacle.solve();
  CatapultA.direction = EAST ;
  CatapultA.distance = 3 ;
  CatapultA.direction = SOUTH ;
  CatapultA.distance = 4 ;
  CatapultA.direction = WEST ;
  CatapultA.distance = 4 ;
}
  
```





## Skill Development

### JavaScript

Constant parameters in functions: I understand how parameters can affect the output of a function.

Changing values (numbers, strings, Booleans): I understand the difference among different types of values and can explain in what situations I might use each.

### Computational Thinking

A3: Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment).

E2: Assessing whether an algorithm does the right thing (functional correctness).

### Computing Progression

YA1: I know that algorithms are implemented on digital devices as programs.

YP2: I can use logical reasoning to predict the behaviour of programs.

PA2: I know that computers need precise instructions.

PP3: I can run, check and change programs.