
Coding with Kids

Douglas Webster

Sep 03, 2022

OPTIONS

1	Just Logic	3
1.1	Key Goals	3
1.2	Options	3
2	Just Code	5
2.1	Key Goals	5
2.2	Options	5
3	Just Hardware	7
3.1	Key Goals	7
3.2	Options	7
4	Hardware and Code	9
4.1	Key Goals	9
4.2	Options	9
5	Contents	11
5.1	Accepted Content Guidelines	11
5.2	Making Suggestions	11
6	Workflow	13
6.1	Overview	13
6.2	Key Scenarios	13

Coding with kids can be a *great* way to not only teach them some skills that will help them with their own lives, but also *create or strengthen* some lasting **bonds between you and them**.

Here is some new content from VS Code.

Here is a **shiny** new note.

Throughout your journey you will both likely learn new **concepts** and **terms** that will expand your horizons and provide hours of enjoyment.

Note: Some kids have problems with screen time and if you leave them unattended with a screen and they have access to things like YouTube or games, they may not stay focused on the task at hand.

There are lots of great options for where to start, and *Just Logic* is a great place to start. Note that the *best starting point* is another way to refer to the same place but give your link some title.

For details about the hardware and code options, see *Options section for hardware and code*

JUST LOGIC

All of the options below involve not screens at all – they are simply physical objects and some problems to be solved.

1.1 Key Goals

The main goals regarding logic that you might pursue with some of the options below are as follows:

- **Understand basic logic**
 - indented item
- Use available options
- Apply problem solving techniques
- Have fun :)

1.2 Options

1.2.1 CodeMaster

1.2.2 Robot Turtles

1.2.3 Primo / Cubetto

JUST CODE

2.1 Key Goals

2.2 Options

2.2.1 Scratch

2.2.2 KidsRugby

2.2.3 Python

Here is some code:

```
def countAdjacent(p, c, r yChange, xChange):
    global board
    adjacentCount = 0

    while True :
        c = c + xChange
        if c < 0 or c > 7:
            return adjacentCount

        r = r + yChange
        if r < 0 or r > 7:
            return adjacentCount

        if board[c][r] == p:
            adjacentCount = adjacentCount + 1
        else:
            return adjacentCount
```

And here is some C# code just in case you wanted to see it:

```
private static string GetMessageFromException(Exception ex)
{
    if (ex == null) return "";
    if (ex.InnerException != null)
    {
        return GetMessageFromException(ex.)
    }
}
```

(continues on next page)

(continued from previous page)

```
}  
  return ex.Message  
}
```

2.2.4 Hopscotch

JUST HARDWARE

3.1 Key Goals

3.2 Options

3.2.1 Arduino

Arduino is an open source electronics platform that is great for both adults and kids.

<https://www.arduino.cc/>

3.2.2 Snap Circuits

[Snap Circuits](#) is a great electronics platform consisting of plastic board and electronic components that you snap together based on project guides to learn basic electronic concepts.

3.2.3 Project Bloks (in development)

Project Bloks is on the web at <https://projectbloks.withgoogle.com/> and is a new project from Google that is still being developed but looks pretty awesome if you have younger kids that want to learn logic and flow.

HARDWARE AND CODE

4.1 Key Goals

4.2 Options

The following table offers some simple comparisons of the different platforms...

Table 1: Comparison

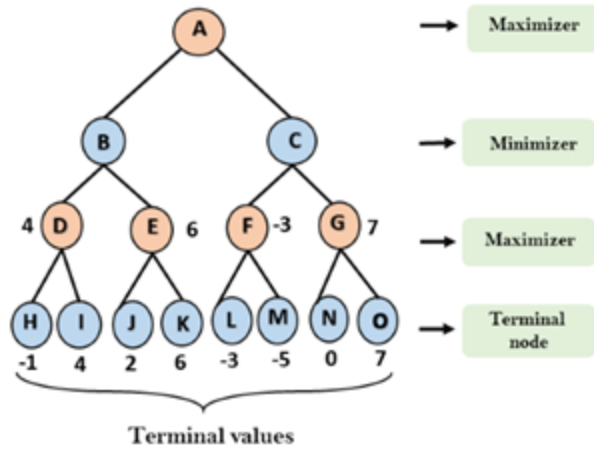
Platform	Self-Contained	Cost	Flexibility	Description
Raspberry Pi	No	\$30	Limitless	Mini computer board with GPIO pins for interfacing and experimentation
Lego Mindstorms	Yes	\$350	Medium	Lego robotics system with motors and sensors. Build a robot, then write logic to move it around and do stuff.

4.2.1 Raspberry Pi (Family)

Here's what a Raspberry Pi 2 and a Sense Hat look like:



With these you can do cool things like program your own version of Connect-4 – you will need to consider how to check for a win by evaluating options as shown in the diagram below:



4.2.2 Lego Mindstorms

CONTENTS

5.1 Accepted Content Guidelines

5.2 Making Suggestions

5.2.1 What to Include

WORKFLOW

6.1 Overview

6.2 Key Scenarios

6.2.1 Add New Page(s)

6.2.2 Editing Existing Pages(s)