Digital Technologies @ Home
Unplugged activities for students

This page is for

This activity is for: Years 3-4, 5-6

Race Up If Mountain!

This activity teaches…

This activity is designed to teach decision making in programming as well as starting to teach students about variables. The game is based on a path up a mountain blocked with a number of smaller 'if mountains' that must be passed to reach the finish. The students must roll the dice and follow the flowchart at each mountain, trying to roll a number that will allow them to pass. The statements have been written in flowcharts that mimic the decision making of computers.

It is targeted towards primary students in years 5-6 and can be expected to take between 1 and 1.5 hours to complete.

An extension for students that are familiar with Python syntax has been included but is by no means expected. In the extension the flowcharts have been translated into Python coded 'If statements' (the name for decision making statements in code). The Python statements are logically identical to the flowcharts and can be used concurrently for students that are keen but inexperienced.

Getting started (read this with your child):

This is a board game where players race up the mountain, it's a game of luck and climbing! First we will build up the board, then we will race to reach the top of the mountain, stopping at smaller mountains along the way. The flowcharts on those mountains will help you decide if you can pass, they are similar to how computers make decisions.

You will need…

- Scissors
- Glue
- A six-sided die (if you don't have any dice, there is one you can make on the board).
- A playing piece for as many players as you have, for example, 2 bottle caps for 2 players.
- Coloured pencils to colour in your board.

Step by step instructions are on the next page

This page is for

Students

Race up If Mountain!

Build your board and race up the mountain!
Climbing over If Statements flowcharts along the way!

Image originally created by Nicole Turner on Pixabay

Step 1

Colour in your board and your dice. You can colour in the mountains as well, but make sure you can still read what they say.

Step 2

Cut out your mountains. Make sure you also cut around the tabs with letters on! This is important! You need them to stick your mountains to the board. Fold your mountains in half. Then you can cut out your die if you need to.

Step 3

Stick your mountains to your board, make sure you stick the mountain with 'A' to the 'A' squares on the board. Once you've built your mountains, you can fold and stick your die.

Step 3

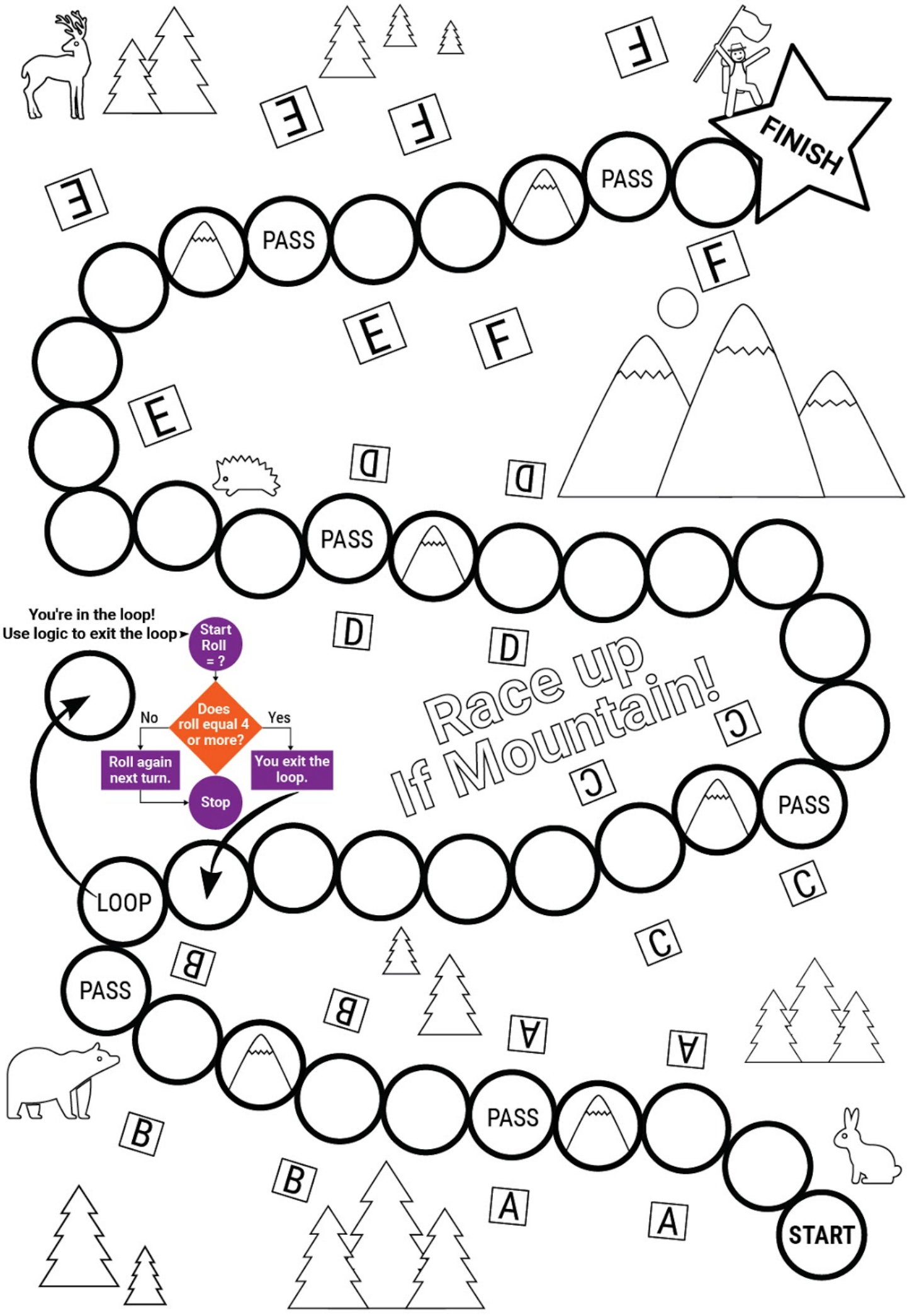
Find players and find a token for each of them.
Once there are at least 2 players, you're ready to go!

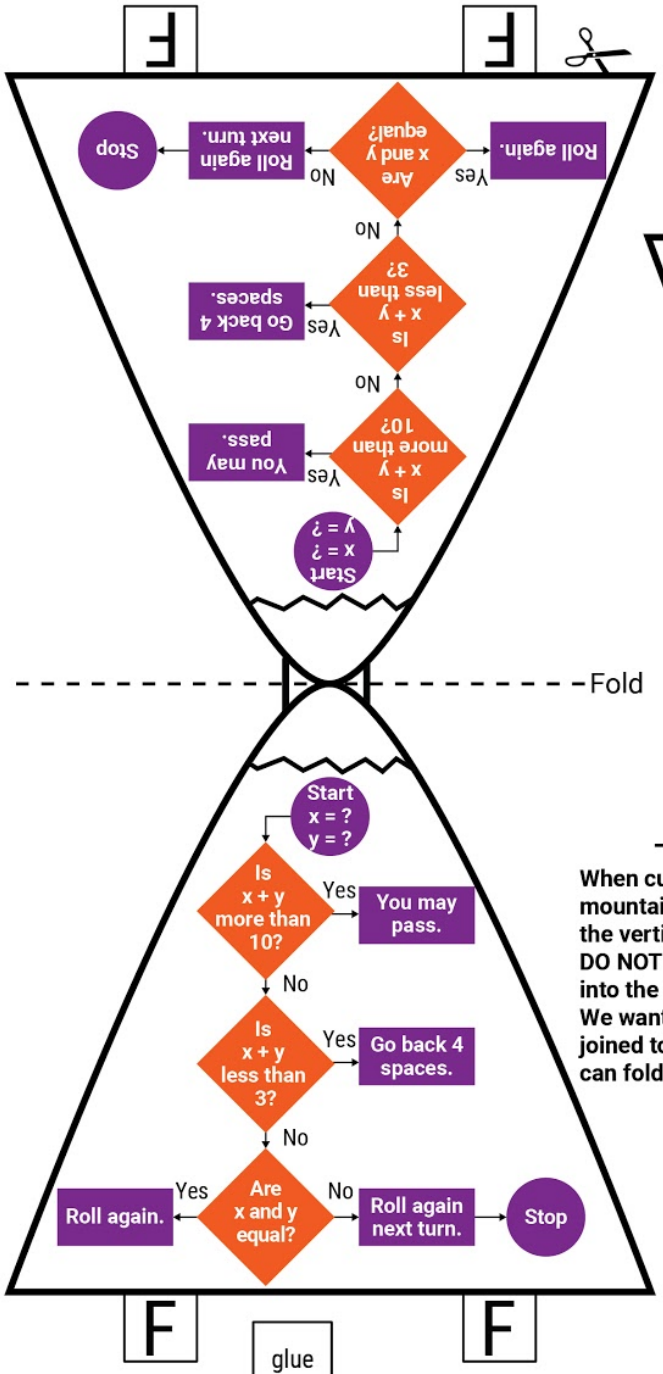
Step 4

Read the rules of the game so you know how to play and start your race!

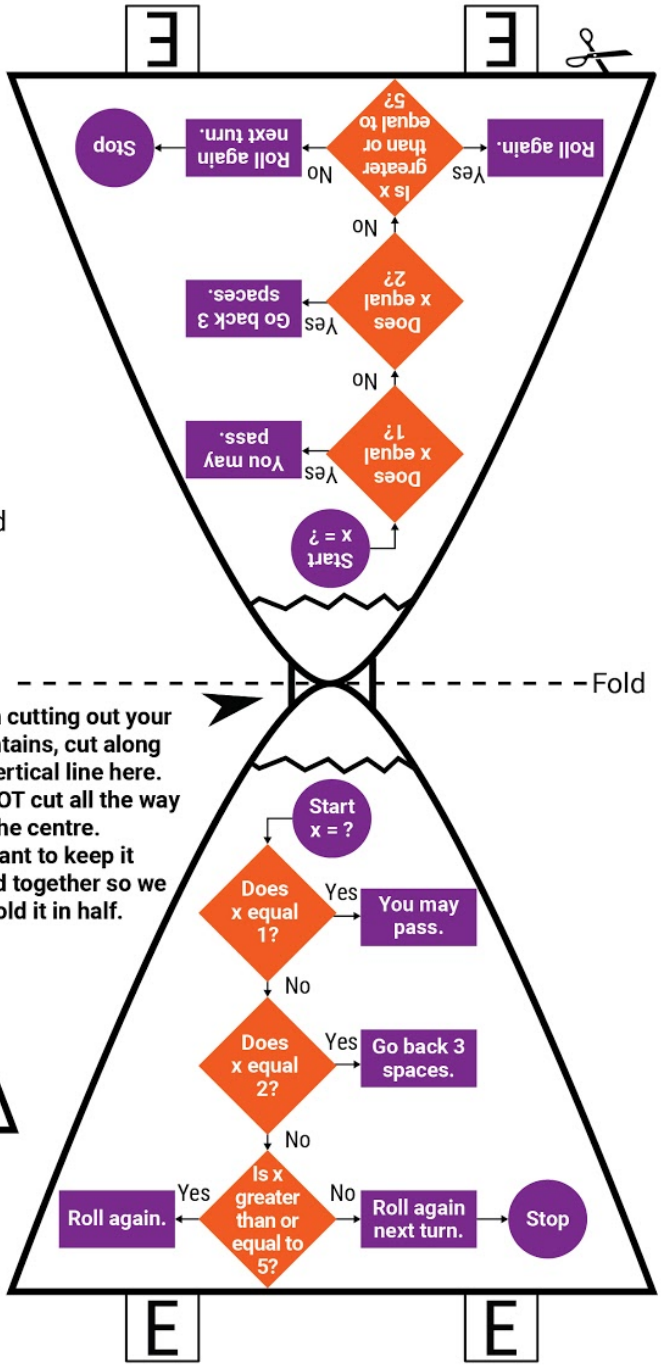
The Rules

- Every player starts at START.
- Take turns to roll the die. Whoever rolls the highest number goes first, second highest goes second etc. If two people roll the same number they have to roll again.
- Move your token as many places as you rolled, rolling a 3 means you move forward 3 places.
- When you land on a mountain you must play the mountain. Roll the die and follow the flowchart. If you pass the mountain, you move to the 'PASS' circle.
- The Loop! If you land on the 'LOOP' circle, you're stuck in a loop! You have to roll a 4 or more to escape!
- The final two mountains use variables. If there is an x variable, roll the die and that number becomes x. If there is a y variable, roll the die again, that roll becomes y. Use those for x and y on the flowchart.
- You must re-roll each variable every turn.
- To win the game, be the first to reach FINISH!


FINISH
PASS
PASS
PASS
PASS
PASS
PASS
You're in the loop!
Use logic to exit the loop
Start
Roll
= ?
Does
roll equal 4
or more?
No
Yes
Roll again
next turn.
You exit the
loop.
Stop
Race up
If Mountain!
LOOP
START

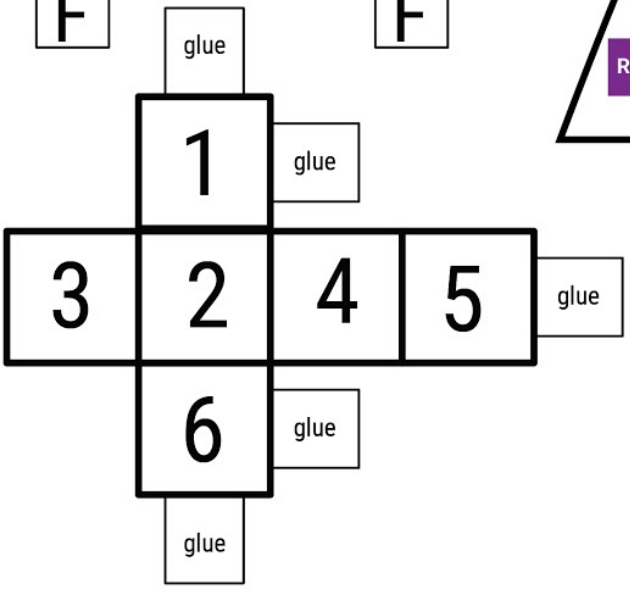


Fold

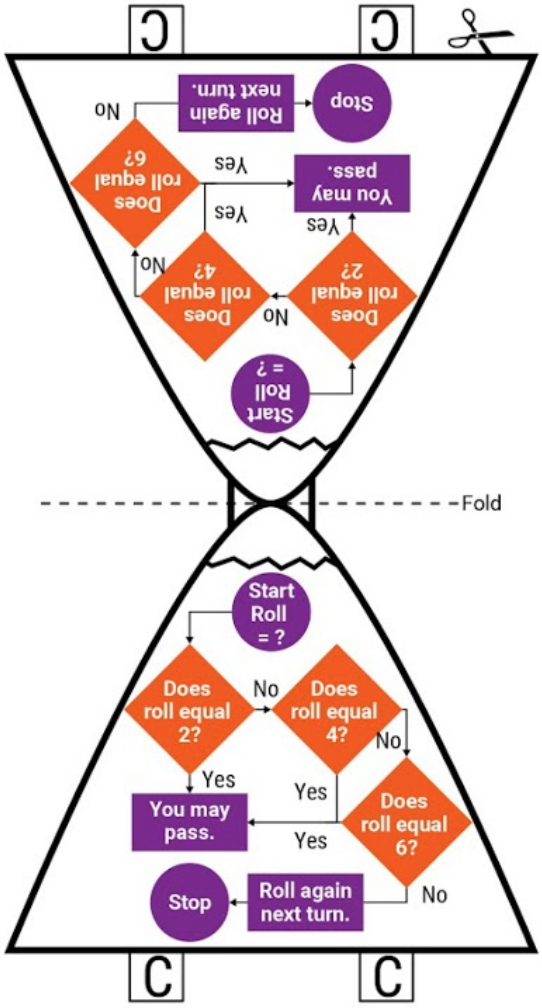


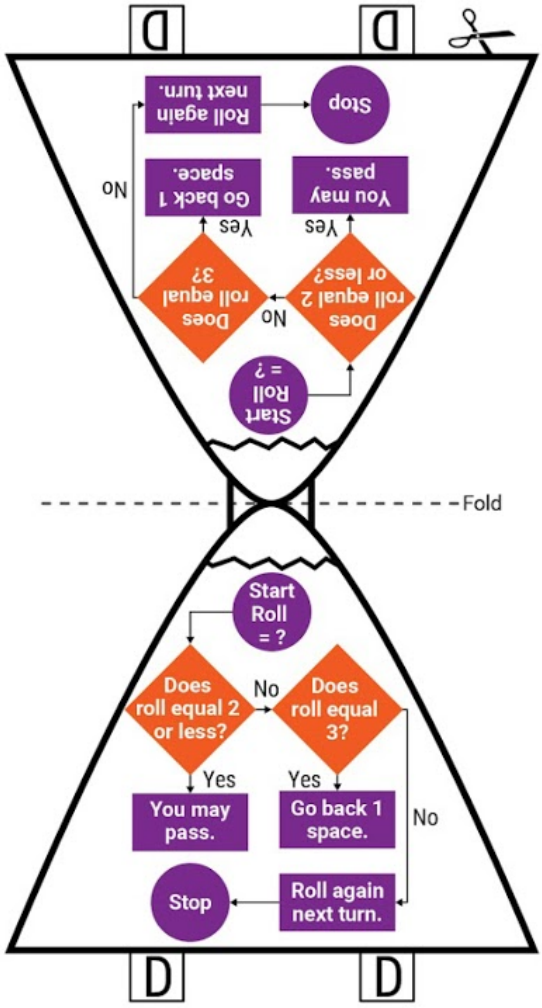
Fold

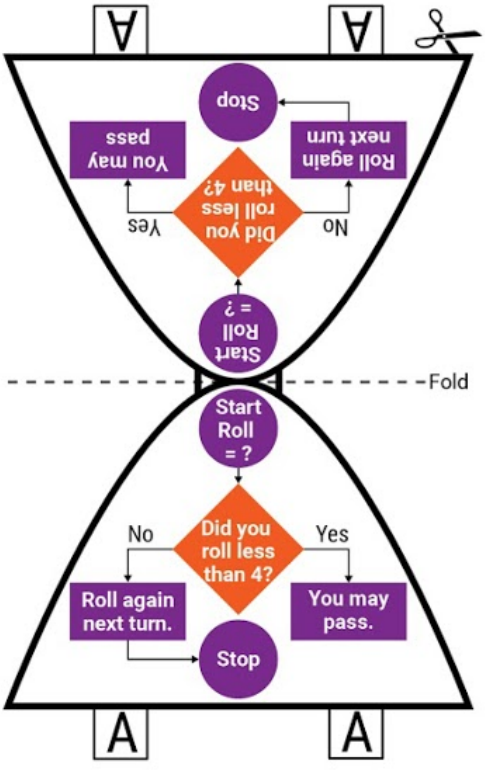
When cutting out your mountains, cut along the vertical line here. DO NOT cut all the way into the centre. We want to keep it joined together so we can fold it in half.

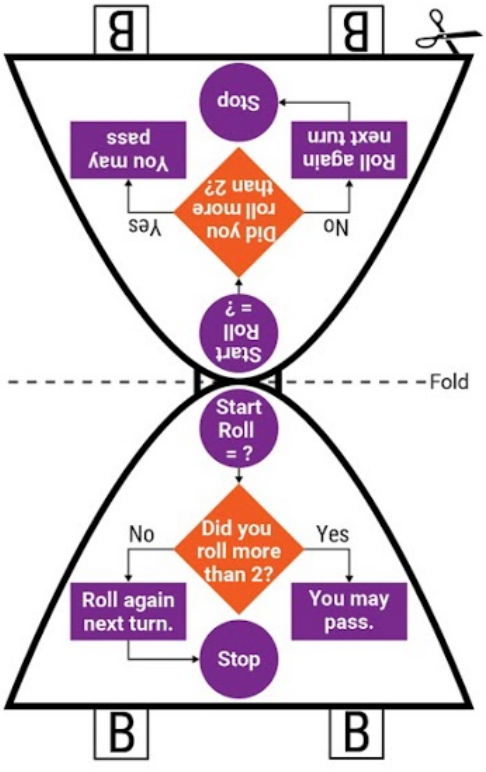


Make sure you keep your tabs when cutting out your mountain! Then dab some glue on each tab and glue them to the board where you see the same letter so they form an arch over the game path.


C
C
Start Roll = ?
Does roll equal 2?
No
Does roll equal 4?
No
Does roll equal 6?
Yes
You may pass.
Yes
Yes
Stop
Roll again next turn.
No
Fold
C
C


D
D
Start Roll = ?
Does roll equal 2 or less?
No
Does roll equal 3?
Yes
You may pass.
Yes
Go back 1 space.
No
Stop
Roll again next turn.
Fold
D
D


A
A
Start Roll = ?
Did you roll less than 4?
No
Roll again next turn.
Yes
You may pass.
Stop
Fold
A
A


B
B
Start Roll = ?
Did you roll more than 2?
No
Roll again next turn
Yes
You may pass
Stop
Fold
B
B

More information for teachers

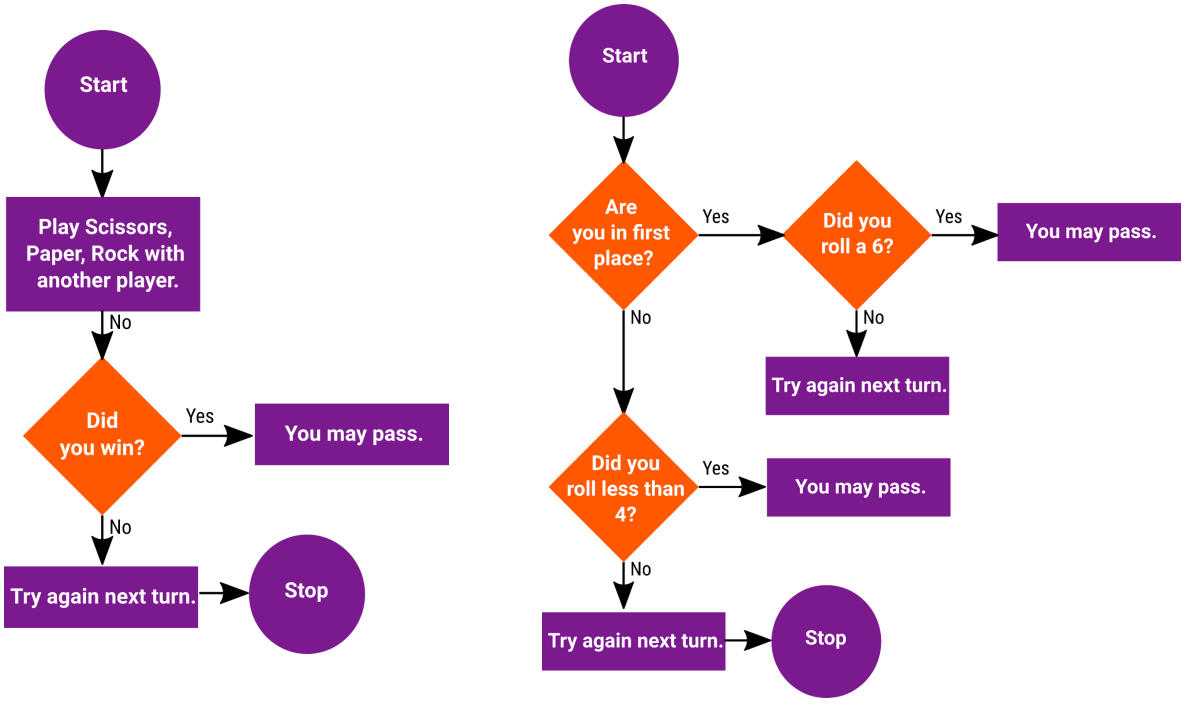
Here are some further activities, online resources, assessment ideas and curriculum references.

Keep the conversation going:

Making your own flowcharts.

Encourage students to come up with their own mountain flowcharts using die rolls as the deciding factor.

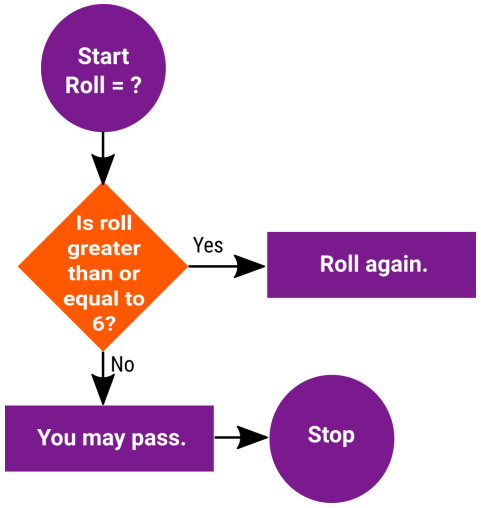
What are other things that your flowcharts can check, aside from the number rolled on the die? For example:



What happens if we accidentally write a flowchart that never lets us pass the mountain?

This is called an 'infinite loop' because you are stuck in the loop for infinity and can't get out.

Ask the student how they would fix it to show that they understand.

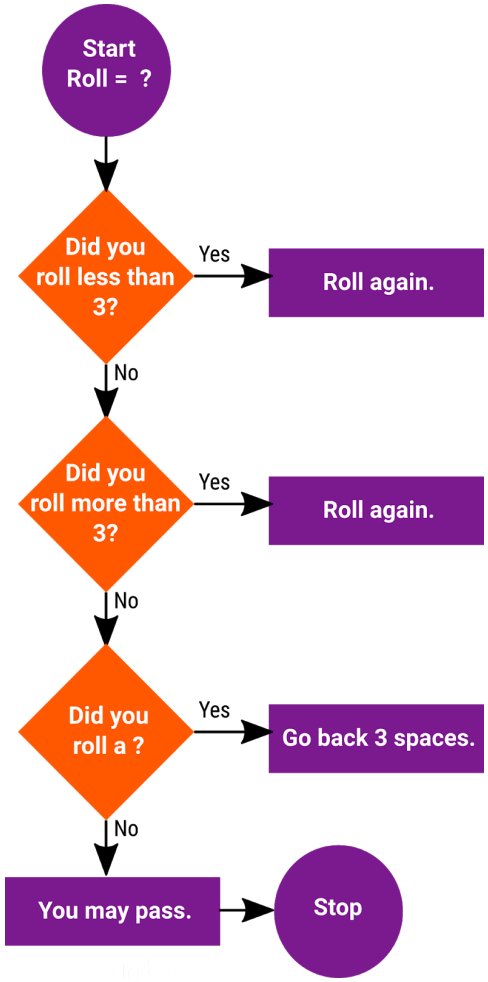


Sometimes they are harder to see.

This infinite loop will never let you pass.

This is because every possible outcome from the dice rolls is caught in a "yes" flow. Since you are always caught by a "yes" flow, you will never follow the "no"s all the way to the "pass" block.

You will also never be able to go back 3 spaces. This is because you will follow the "greater than 3 → yes" flow to roll again every time you roll a 6. This means you will not be able to reach the "equal 6 → yes" flow.



This page is for

Teachers

Thinking about mountain flowcharts and difficulty

How might we make the flowcharts of the last 2 mountains easier or harder to pass? Think about how much more likely it is to roll a number greater than 3 than it is to roll a 6.

For teachers creating a portfolio or learning or considering this task for assessment:

Ask students to design their own mountains.

The first mountains should be small flowcharts with easy odds to pass. They should get progressively trickier to pass. You can decide if they must include variables.

- Make sure the flowcharts are logically exhaustive (expanded below).
- Ask the students to explain why their last mountain is harder to pass than their first. This should demonstrate an understanding that the odds of rolling any number higher than 3 is higher than the odds of rolling a single number.

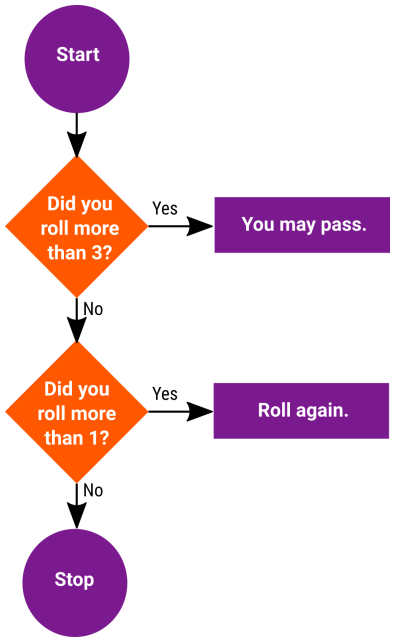
Watch out! Common if statement pitfalls

It is important to make the if statement logic *exhaustive*. Don't miss any of the possible outcomes (dice numbers). For example:

If we roll 1 in the above example then neither of the conditions apply.

What is the player meant to do? They can't roll again or pass, so they are stuck!

The solution is to add a final instruction block before the stop circle that gives an instruction to a player that they should follow if no other flows apply to them.



This page is for

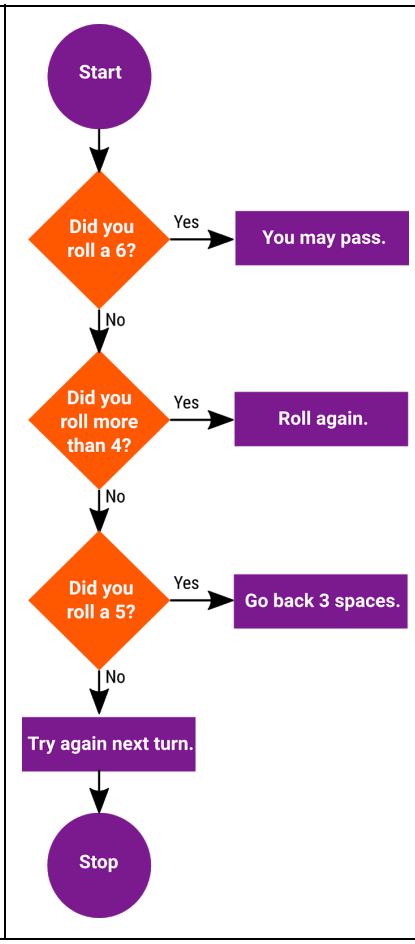
Teachers

It is important that students also understand the *sequential* nature of the flow charts, that they work from top to bottom and that you cannot jump to different diamonds.

In this example, if a player rolls a 5, they will never actually go back 3 spaces.

This is because they would follow the "more than 4 → yes" flow and roll again every time.

To fix this, you would need to swap the "more than 4" and "roll a 5" diamonds.



Helping to Understand Variables

A table is also a good way to keep track of the variables in the last 2 mountains. The last 2 ask you to "*roll for x*" and "*roll for x and y*". Here is an example table you can use to keep track of the variables. You can use this table as a reference when following the flowcharts, for example to work out (x+y).

	What did you roll?	Fill in the number
Variable x		x =
Variable y		y =

Linking it back to the Australian Curriculum:Digital Technologies

Algorithms - Design, modify and follow simple algorithms involving sequences of steps, branching, and iteration (repetition) (ACTDIP019 - see cmp.ac/algorithms) (5-6).
Refer to aca.edu.au/curriculum for more curriculum information.

This page is for

Parents and carers

Extension

Choose if you want to print this for your kids or keep it to yourself!

The flow charts can all be translated into Python Syntax if statements. We have provided these at the very end of the activity. A key component of the Python version is ***comparison operators***, a table below can be provided to help them master these.

Comparison operators conversion table

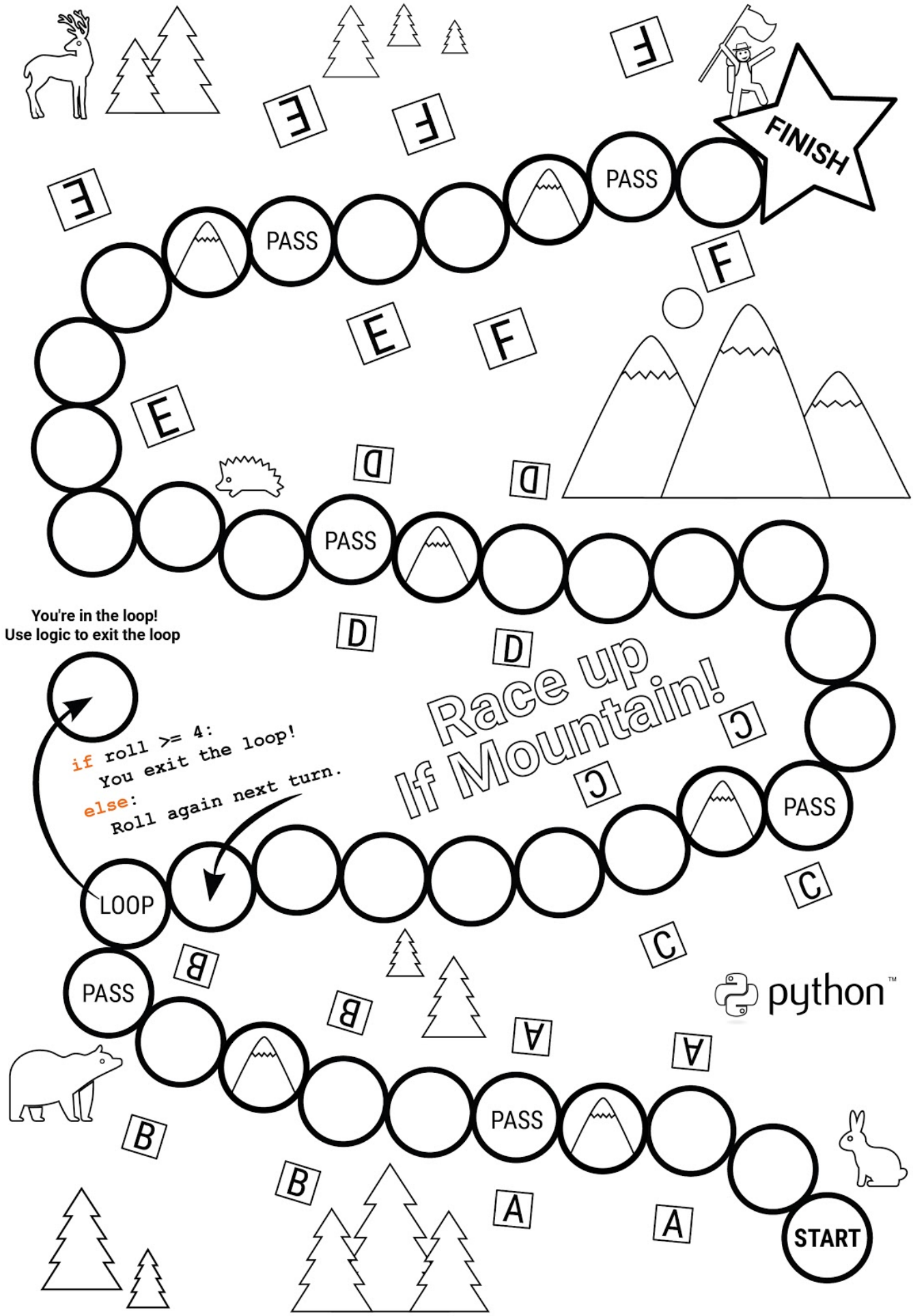
Operator	In a sentence
a < b	a is **less than** b
a <= b	a is **less than or equal to** b
a > b	a is **more than** b
a >= b	a is **more than or equal to** b
a == b	a is **equal to** b

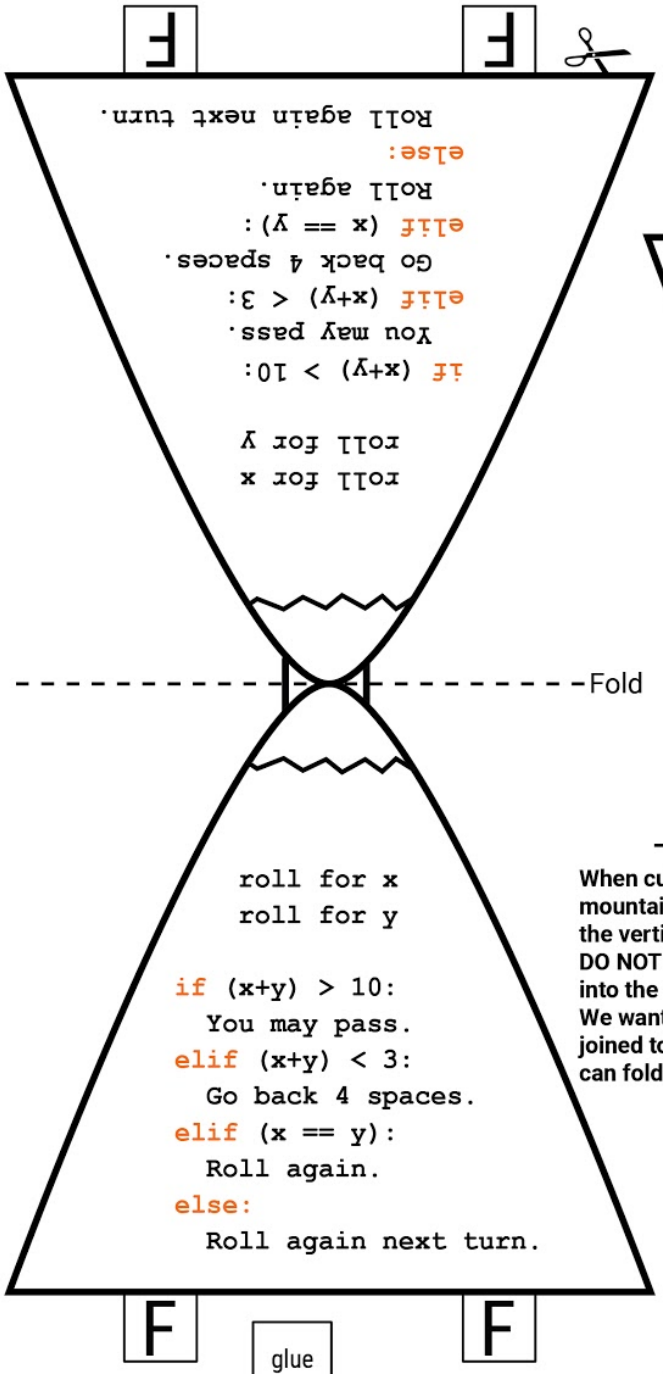
If your child has not learned about if statements and variables before,a good introduction to teaching if statements can be found at:
https://medium.com/groklearning/programming-in-primary-school-introducing-if-statements-64875db05614

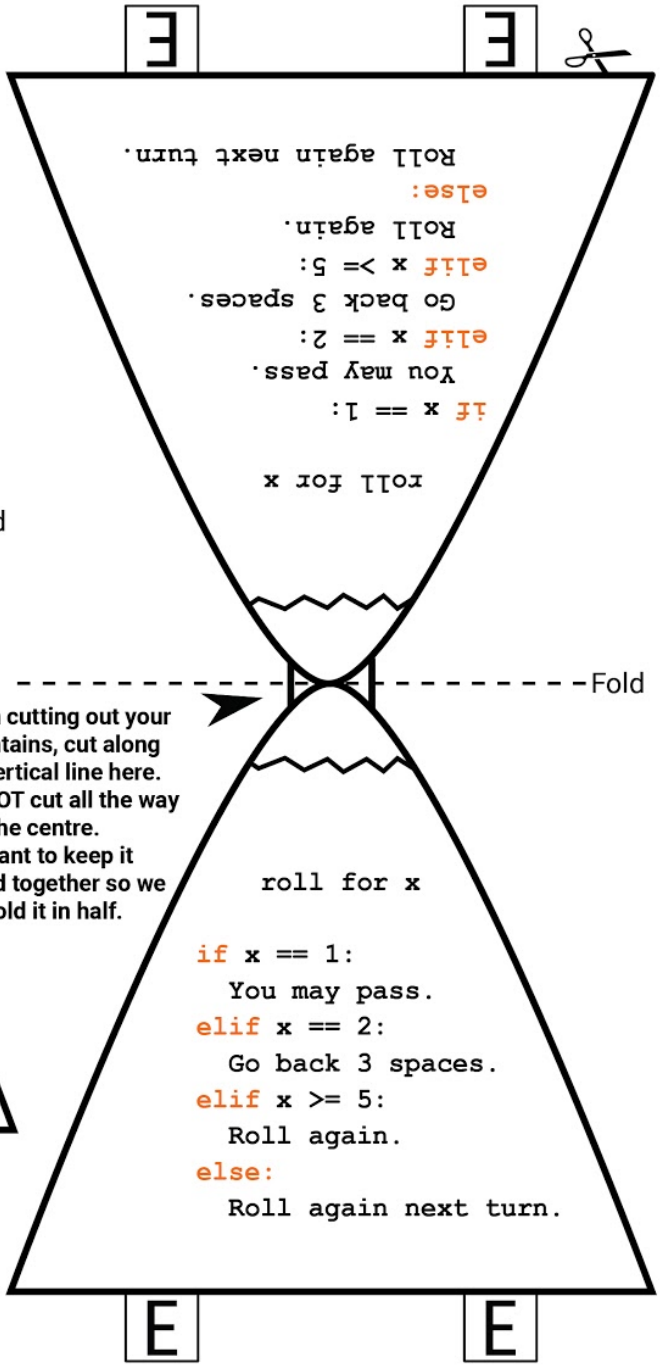
Is it also worth making sure the students can understand what the if statement conditions mean. Have them use the syntax and flow charts to write out sentences for the conditions, to help with their understanding.

If Statement	What does it mean in a sentence?	What numbers will work?
If roll > 3 You may pass	If you roll a number that is greater than 3, you can pass.	4, 5, 6
Else: Roll again	If you **do not** roll a number that is greater than 3, you have to roll again.	1, 2, 3

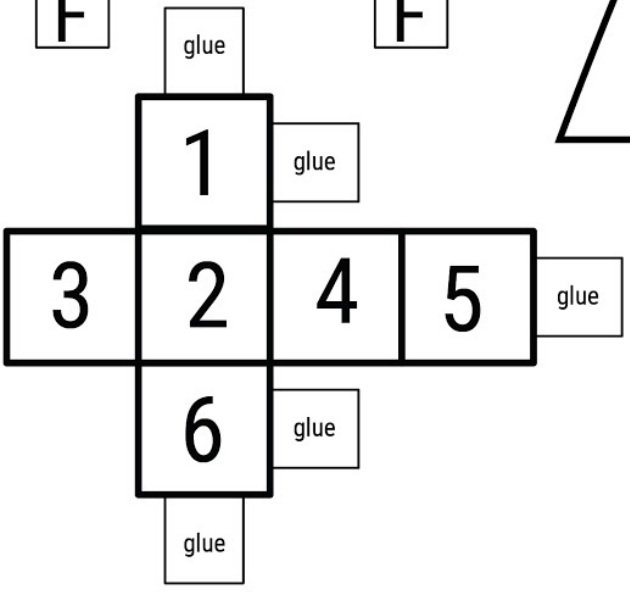
You should encourage them to create a table each time then get stuck with a mountain written in syntax. Using the tables can help students to understand the options available at each mountain and can be phased out as their confidence improves.


FINISH
PASS
PASS
PASS
PASS
PASS
PASS
You're in the loop!
Use logic to exit the loop
if roll >= 4:
You exit the loop!
else:
Roll again next turn.
Race up
If Mountain!
LOOP
START
python™

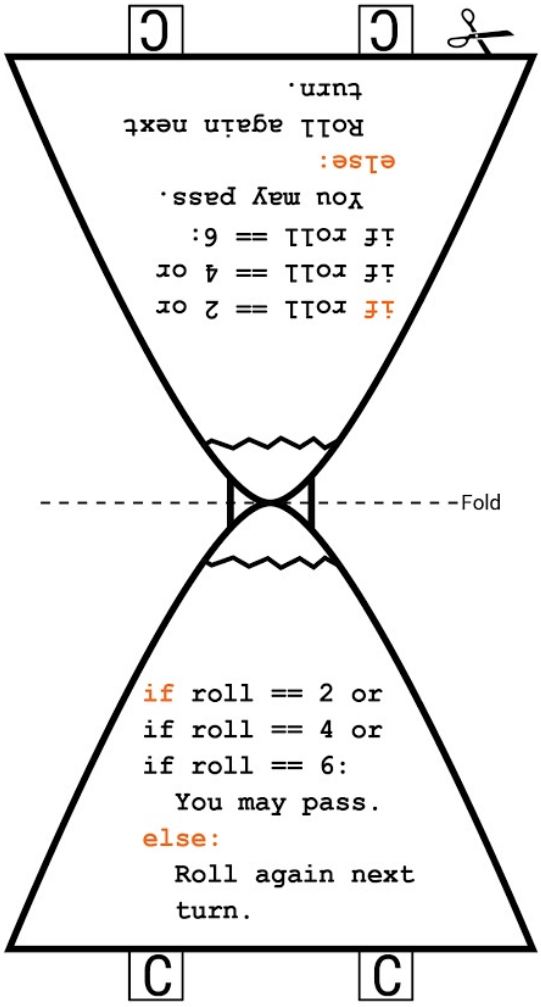


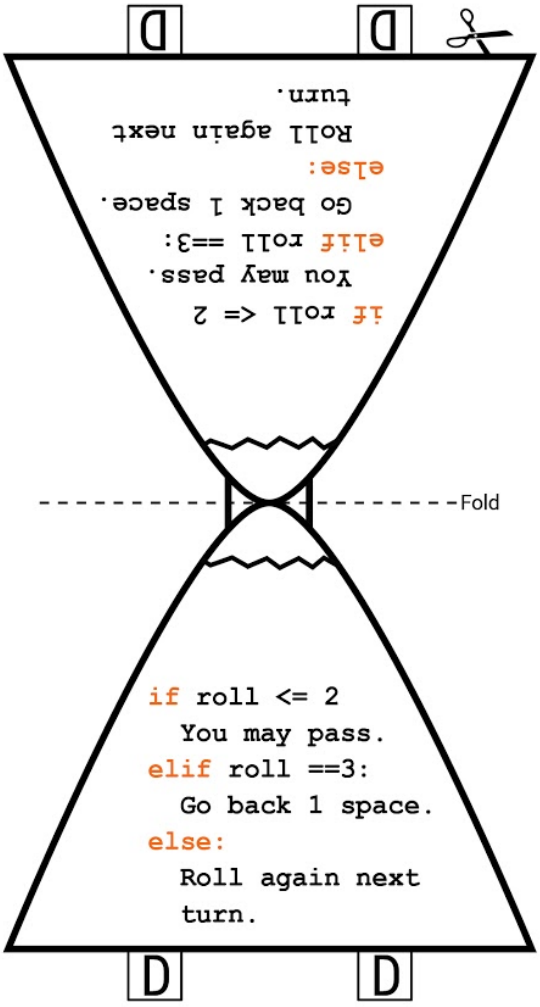


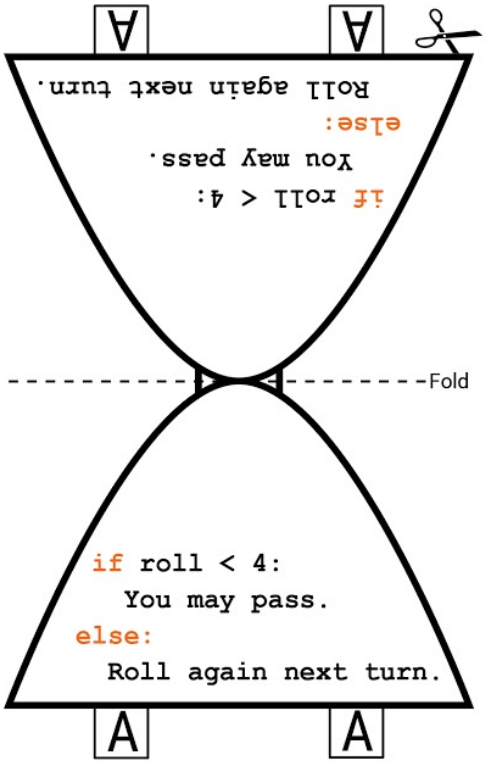
When cutting out your mountains, cut along the vertical line here. DO NOT cut all the way into the centre. We want to keep it joined together so we can fold it in half.

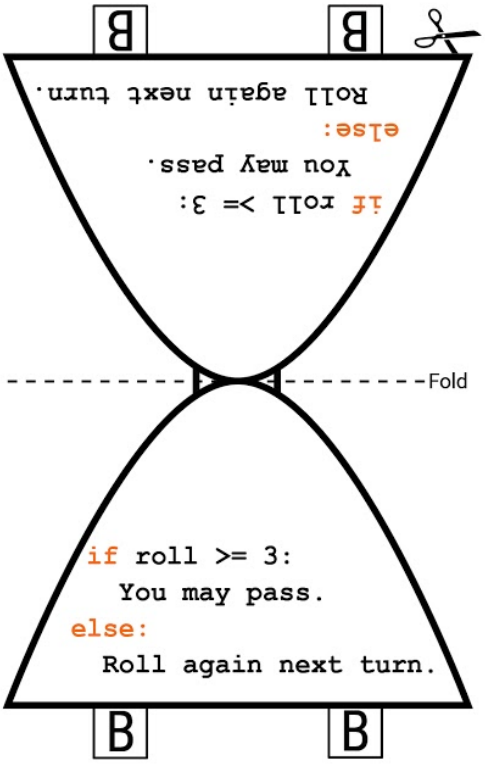


Make sure you keep your tabs when cutting out your mountain! Then dab some glue on each tab and glue them to the board where you see the same letter so they form an arch over the game path.


if roll == 2 or
if roll == 4 or
if roll == 6:
You may pass.
else:
Roll again next
turn.
Fold
C
C


if roll <= 2
You may pass.
elif roll ==3:
Go back 1 space.
else:
Roll again next
turn.
Fold
D
D


if roll < 4:
You may pass.
else:
Roll again next turn.
Fold
A
A


if roll >= 3:
You may pass.
else:
Roll again next turn.
Fold
B
B