

# Karel Project #17

## TRUE AND FALSE

Requiring **boolean operations** can create opportunities for some **clever mazes**.

### Before You Start

Go to the [Creative Suite](#), open a new Karel project, and save it in the folder [course-karel-projects/](#) as [true-and-false](#).

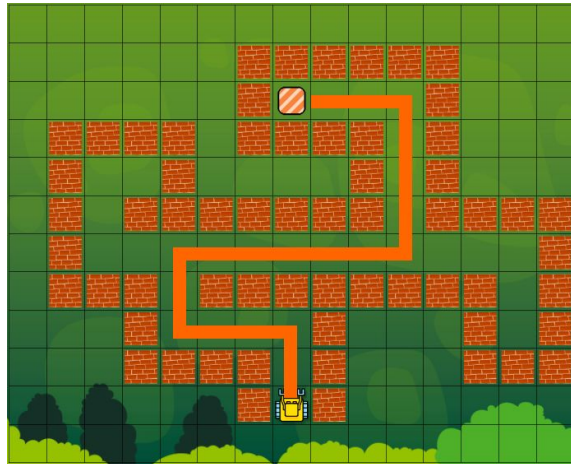
### Goal: Create a Maze with Binary Choices

To [present the player with binary choices](#), you will:

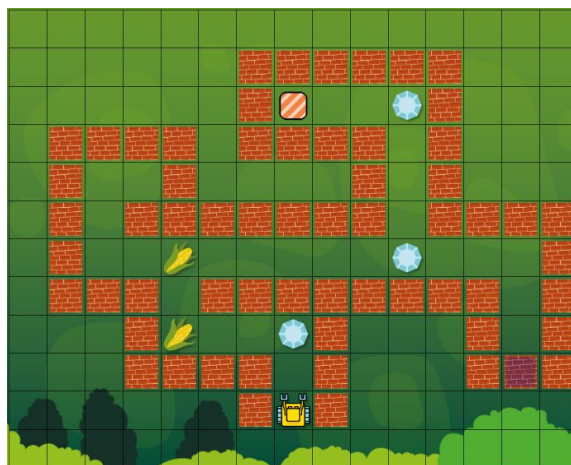
1. Require [true](#) and [false](#) in [Goals](#).
2. Create a [maze with binary choices](#).

## Binary Decisions

A binary decision can only have two outcomes. For the purpose of Karel mazes, we need to consider this in terms of **directions the player will move**.



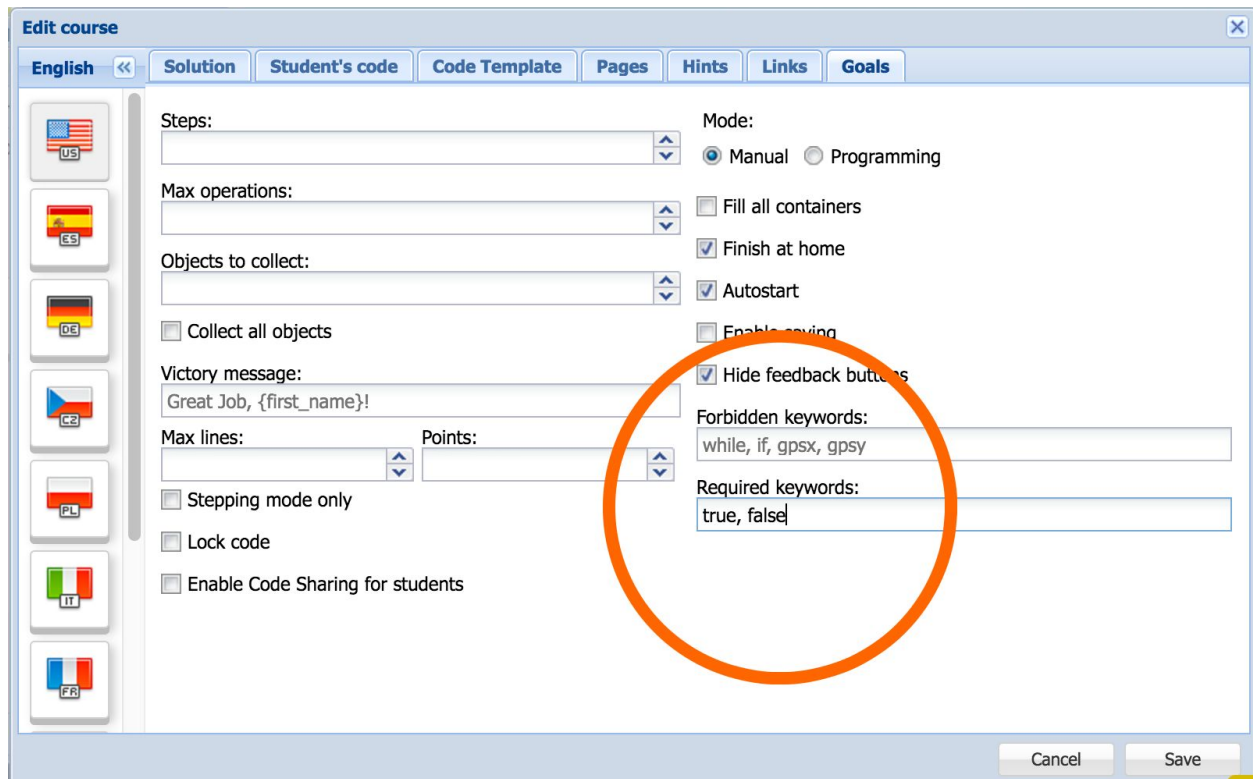
In this maze, the decisions for the player boil down to **left turns and right turns**. When paired with a few Objects, this maze becomes a suitable **programming challenge**.



In this example, **left turns and right turns** were used as **outcomes of a binary decision** based on the type of **Object**.

## Step 1: Require True and False

In this case, you want to force anyone who plays your maze to use **boolean operations** to solve your maze. This is easily done by using the **Goals** menu.



**True and false** are the two keywords to make a **boolean operation**. Now that you have forced the player to use true or false, you need to provide them with challenges that involve **binary decisions**.

## Step 2: Create A Maze with Binary Decisions.

To review, you need to build a maze that **forces the player to use true and false**. This is accomplished by using **binary decisions** to shape your maze. In a game you can not force a player to do anything, but you can **persuade** the player to make decisions with good **rules**. The player might come up with a solution that you had not considered. That's the fun of **game design!**



## Project Checklist

Your Project will be finished when:

1. You have saved the project in the folder **course-karel-projects/**.
2. You created a **maze with binary choices**.
3. You used **Objects and Obstacles** to lead players to the goal.

