Manual Scratch Unplugged

Scratch Unplugged is an activity designed by the CodesCool team. The goal of CodesCool is to get as many pupils as possible in primary education to code without anyone dropped by the wayside! Always free, always fun! www.codescool.be

Scratch?

With Scratch you can program your own games, tell stories, set up your own robot, ... In short: you write a computer code with it. The great thing is that you cannot type errors with it: it connects different encryption codes together.

Unplugged activities

Those activities highlight your coding skills without using a computer or a robot. You program as it were your teammate! One member of the team plays the robot, the others act like a computer and combine their coding powers!

Note: the robot cannot think, he only performs!

What do you need?

First of all, print the pdf twice. This file contains all figures you need. Preferably print it in colours, but black and white is no problem either.

Laminating makes the game endurable: you can play it endlessly.

Then cut out all figures and building blocks.

Provide two non-permanent markers.

Rope (knitting yarn, paper tape,...) to divide the playing field.

The preparations

You divide your playing field using the rope or paper tape: it becomes a sort of draughtboard on the floor. The more divisions you make, the more difficult the game will be. For example, you can start by making a field of 8 rows and 8 columns.

(Tip: do you have a floor with square tiles? This is useful since you can use the tiles as playing ground and you only have to tape the sidelines!)

The game manager spreads all 'walls' and 'bombs' arbitrary over the playing field.
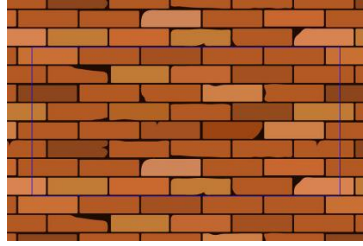
The CodesCool robot is the starting point of every team. Both robots have to start on a square on their sideline, each on one side. After the instructor placed all 'walls' and 'bombs', each team may choose the square on their line as their starting point.

The goal

The goals of the game is to cross the playing field with the CodesCool robot as fast as possible and to reach the starting point of your opponent, without walking on a bomb or running up against a wall.

How?

One team player is the robot: He or she carries out the instruction on the playing field given by the other team players (the 'computers') without hesitation or opposition.

Each round, the 'computers' combine some of their Scratch cards to one single algorithm. When both teams are ready, the instructor gives the go-ahead and both robots can start carrying out the instructions of their own team. Each algorithm starts with one of the following blocks:

We give an example:
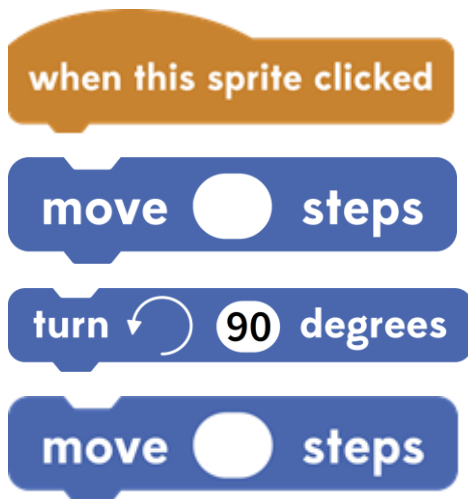
The execution goes as follows:

When the 'computer' clicks on the 'robot', then the latter takes the number of steps the 'computers' have filled in in the blank field with the erasable markers, turns 90 degrees to the right and takes again x steps forward.

Once the two robots have executed their algorithm, a new round starts and the computers can start working again to combine Scratch cards into a new algorithm. They use as many or as little Scratch cards as they want.

Both teams are ready? Then every robot executes again his/her new algorithm.

The game ends when the 'robot' reaches the starting point of the other robot.

What you still need to know

- Did you walk on a bomb? Then you have to go back to your starting point. Walking on a bomb is 'deadly'.

- Did you run up against a wall, then you stay where you are, but you can still perform the rest of the algorithm.

- Once the algorithm is being executed, it cannot be changed anymore.

- All cards can (but should not) be used, but only once per round.

- On the cards with a white area, the 'computers' may write down a value: the variable. Therefore, they use the erasable pen so that the cards can be used again or the rectangular blue block scan be used.

The blocks

This is the starting sign for your robot. In the white area, the 'computers' write what they think is a jolly start: a pat on the back, clapping, a whistle,... . Do not hesitate to be original!

when this sprite clicked

This one has the same function as the button above. To start the robot, "click" on him or her.

when 🚩 clicked

This one has the same function as the button above. To start the robot, "click" on the little flag.

You always have to start the algorithm using one of the three blocks above.

if ⬜ then

In the "if...then" block, you don't write anything but you put one of the light blue blocks on it ("touch colour?" or "touch obstacle?"). On the inside, you can put any block you want that fits in it (which has a tenon and a mortise).
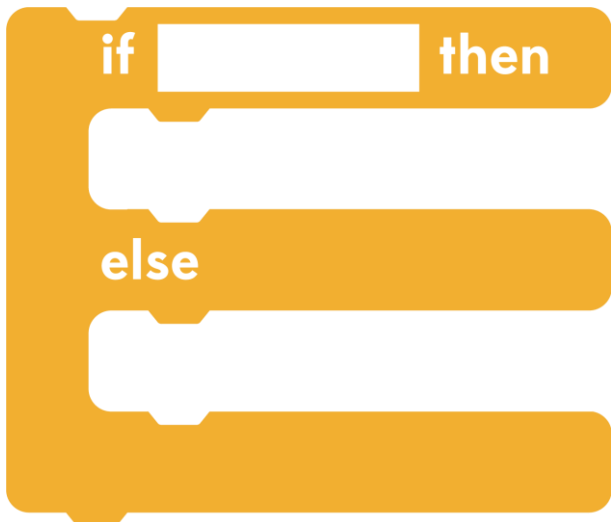
This is the same block as above, but with an extra: If the first block doesn't need to be performed, then you have to perform the second.

Every block with a tenon and a mortise fits and should be executed endlessly (till the robot ends up against a wall or on the side of the playing field, reached the finish or walked on a bomb!)
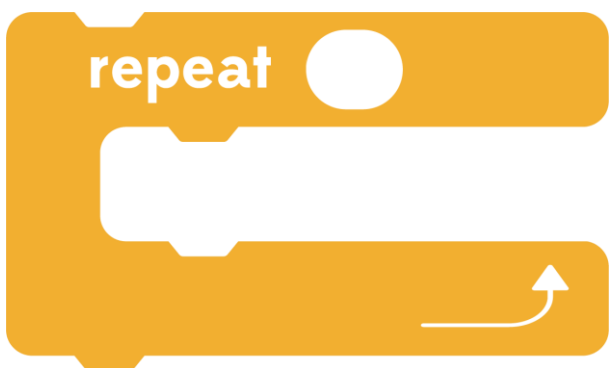
This block has exactly the same function as the block above, but the 'computers' write themselves a number in the circle with the pen. This indicates the number of repetitions!

This block also allows to repeat, but only the light blue blocks fit in the rectangular area ("I touch colour?" or "I touch obstacle?").

The robot keeps walking straight ahead till he/she reaches the side. There, the robot stops and turns 90 degrees to the right.

In the white circle, the 'computers' write a multiple of 90°. In some of these blocks, "90" is already filled in. Pay attention to the turning direction: this can be to the left or to the right!

In the white circle, the 'computers' may write an integer. A robot only walks in a straight line and forward. If the robot hits a wall or an obstacle before all steps have been taken, he/she may stop and move on to the next line of the algorithm.

In the "point towards" block, the 'computers' may write something of their own choice: the nearest bomb, the blackboard, a colored tile, the door, ... Note: the 'robot' can only make a rotational movement of right angles. It is impossible to stand obliquely in relation to the playing field.

A totally unnecessary block that lets your robot say something when executing the algorithm. In the rectangle you can write what you want; as long as it stays tidy!

An equally redundant block like that above. The only difference is that the 'computers' in the white rectangle may write a "sound-making object". The robot then imitates it without complaining!

Let's play!

-The playing field is demarcated with the rope or tape, the number of subdivisions is agreed. The first time you play, 8 rows and 8 columns is a safe choice.

-The participants are divided into 2 groups. Each group determines who is the 'robot', the others are all 'computers'. There is one game manager. (Tip: it can be nice to pitchfork the most dominant student to 'robot': he or she may not say anything, let alone critically approach the orders received! A lesson in humility ;-))

-The game manager randomly puts the bombs, colored tiles and the bricks over the playing field. Both teams place their robot icon on a square of their baseline of the playing field. Think carefully where you want to start!

-The game manager gives the go-ahead and both teams get to work with (a number of) their cards to create an algorithm: a succession of commands for their robot.

- After 2 minutes the game manager says "stop!": both teams stop programming.

-The robot of the team that has the least number of building blocks in the algorithm can start with its execution: all commands are executed from top to bottom.

-The second team then executes their algorithm step by step.

-Then, the second round starts: both teams start to build a new algorithm with the aim to bring their robot closer to the finish.

-The team whose robot first arrives at the starting point of the other robot wins! This can already happen after the first round, but also after round 2, round 3, ...

Levels

Level 1: the yellow building blocks are not used yet. All basic rules are valid.

Level 2: the yellow building blocks may be used (but this is not obligatory!). All basic rules are valid.

Level 3: at least one yellow building block must be used per algorithm.

Level 4: each robot is allowed to take a bomb that is in an adjacent box and leave it at a different location in an adjacent box later on. This, of course, with the intention of stabbing opponents in the wheels.

Level 5: The game manager randomly lays the Bitcoins over the entire playing field. There is a starting point, but no end point. The aim of the game is to collect as many Bitcoins as possible. You can collect these by stepping over or standing on them. The game ends when every Bitcoin is collected.