Cracking the code: Robots in the classroom

Video transcript for ‘Cracking the code: Robots in the classroom’

Minna Kane: Have you ever wanted to own your own robot? Robots are really cool, but they’re useless if you can’t tell them what you want them to do, in a way that they’ll understand. So I’m here at the Digital Schoolhouse with some pupils from Marish Primary School and they’re going to show me how they program their own robots.

We’ve got a map of a town, full of buildings like hairdressers, florists and a gym. Our car is going to start next to the sports shop at A and then drive through the town to the bakers at B.

Teacher: FD, RT90, FD.

Minna Kane: First of all, we’ve got to plan our route, breaking it down into a sequence of step by step instructions. We can get around the whole town with just three instructions – move forward, turn left 90 degrees, and turn right 90 degrees. So that our car follows the road through the town avoiding all buildings.

Teacher: FD, FD, Gone.

Minna Kane: Now we’re re-creating the town on our computer, using different pictures to represent the shops and buildings.

Child 1: We’re making this like a town so the car can move around. We’re trying to match these pictures with those pictures over there.

Minna Kane: OK. So all these objects represent kind of shops in the town, is that it?

Child 1: Yes.

Child 2: So we go to gym, press gym.

Minna Kane: So you’ve just added gym to the objects of things that now are on your map?

Child 2: Yes, so it’ll make it more interesting, because then the robotic car can go, like, to lots of different places like everyday life.

Minna Kane: Oh cool.

Cracking the code: Robots in the classroom

Video transcript for ‘Cracking the code: Robots in the classroom’

Minna Kane: Now we create a procedure called drive.

A procedure is the list of instructions that our car is going to follow. It will only do exactly what we tell it so we’ve got to make sure we’ve got all the instructions right, otherwise the car will go the wrong way. So can you tell me about the commands?

Child 4: Well I’m telling it to do forward, forward, right turn 90 degrees.

Minna Kane: So the FD means forward and then the RT90 means right turn 90, so that 90 is the degrees you want your car to turn?

Child 4: Yes.

Minna Kane: These simple instructions are the building blocks of our code and with the right sequence of instructions we can make the robot travel any route we like. So where will your car end up then?

Child 4: Next to the bakery.

Minna Kane: Ok. Fingers crossed, that’s if everything works out.

An important part of programming is to test the code that we’ve written. Some of the groups have decided to test their procedures on the computer before they try it out with their actual robot. That’s a good idea, because it’s going to be much quicker to change it now rather than later.

Looks like the car is crashing into the gym. Might need to check the code guys!

Once we’ve tested it, we need to connect up our car and download the procedure, so our code can tell the robot what to do.

And now, the moment of truth, have we got it right?

We’ve made it past the first bend. Hang on, it’s all gone wrong on this corner. We should have turned right 90 degrees but we’re going back the way we came. How did that happen? And now we’ve crashed into the cake shop. That’s what happens when you don’t test your code properly.

Cracking the code: Robots in the classroom

Video transcript for ‘Cracking the code: Robots in the classroom’

Minna Kane: Looking good so far.

So close, but they’ve gone forward again. They’ve missed a turn, now they’re not even on the map, and into the crash barrier.

Child 4: I was meant to do an RE90 and then move forward twice and then do an LE 90.

Minna Kane: Ok, so it was the right and left you got wrong basically. Oh that’s such a shame guys. Do you think it’s going to work?

Child 4: Yes.

Minna Kane: Do you guys think it’s going to work?

Children: Yes!

Minna Kane: Ok, let’s go. Fingers crossed.

Tension’s mounting.

Brilliant, we’ve made it. That’s great guys. We’re next to the baker, just how we wanted to be, cool.

It looks like it’s going to be easy just to write a set of instructions doesn’t it, but actually it’s quite hard. But you need those instructions to be precise so that your robot knows exactly where to go – that step by step process.

When you’re programming, you often have to find and fix mistakes until your code does exactly what you want. We call mistakes bugs and getting rid of the bugs is all part of the fun.