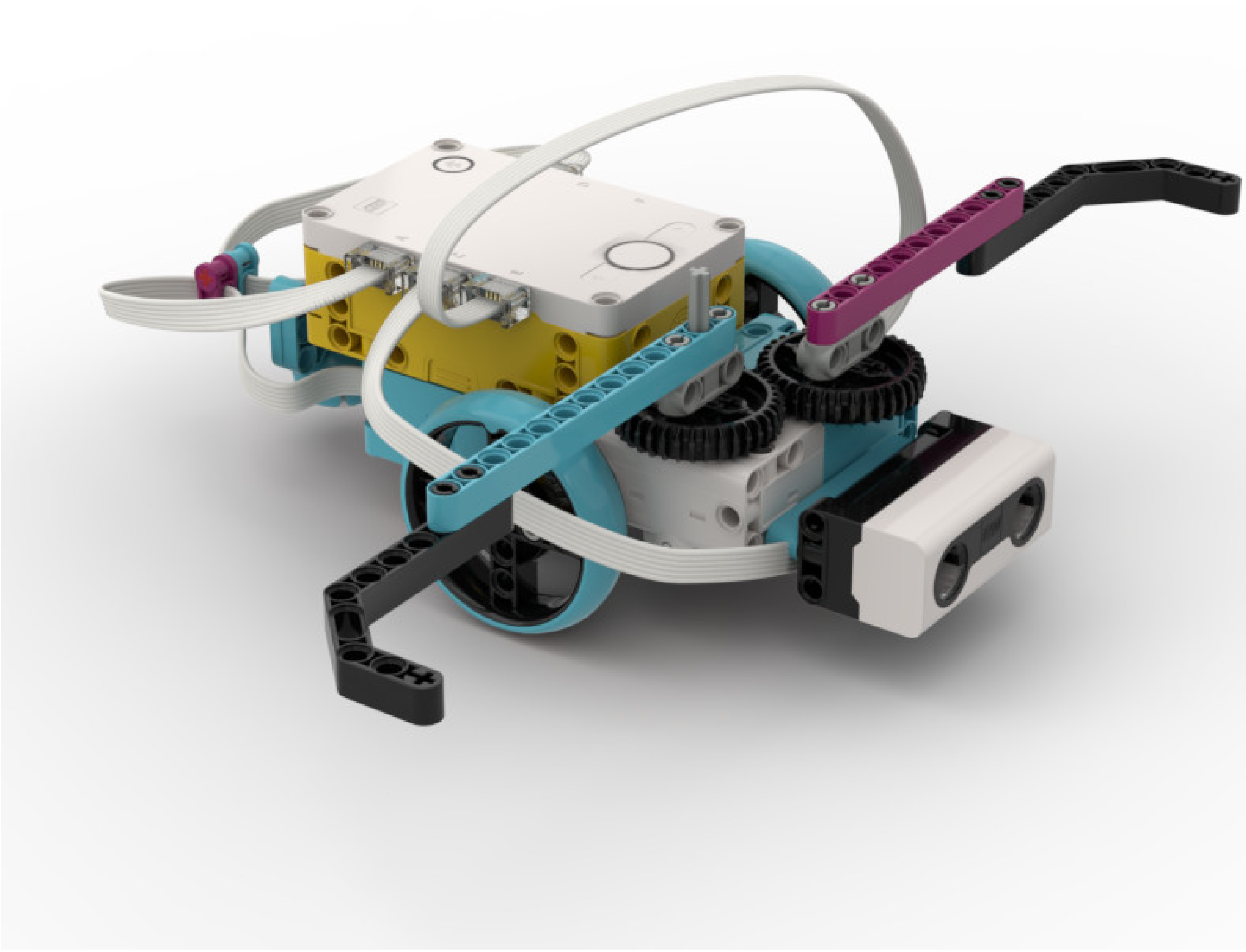
Classroom Activities for the Busy Teacher: SPIKE Prime



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Contents

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Chapter 1: Introduction

This book is a guide for teachers implementing a robotics unit in the classroom. It is aimed at Primary and Middle years schooling (ages 7 - 15) but the wide range of activities can be adapted to suit older or younger students. The book is based around a single robot, "SPYKEE", which is used in all activities. This approach is valuable in resource limited classrooms, as it allows the teacher to work with a 'standard' robot, rather than using valuable classroom time building and breaking down robots each lesson. The SPYKEE design can be found at the back of the book, as well as being freely available online – www.damienkee.com. Please send me an email and let me know if you are using the design!

All activities are based around the 45678 LEGO SPIKE Prime kit available from LEGO Education.

It is assumed that the teacher has a basic knowledge of how to open the SPIKE Prime programming environment and how to download a program to the SPIKE hub. Please see the excellent tutorials built into the SPIKE software environment for more information.

The book is divided into sections that follow a 10 week plan, although this can be modified to suit the needs of the teacher. The first 6 weeks takes students through a series of activities, progressively exposing them to new aspects of the SPIKE programming environment. Following is a set of open-ended challenges from which teachers may pick and choose to suit their class.

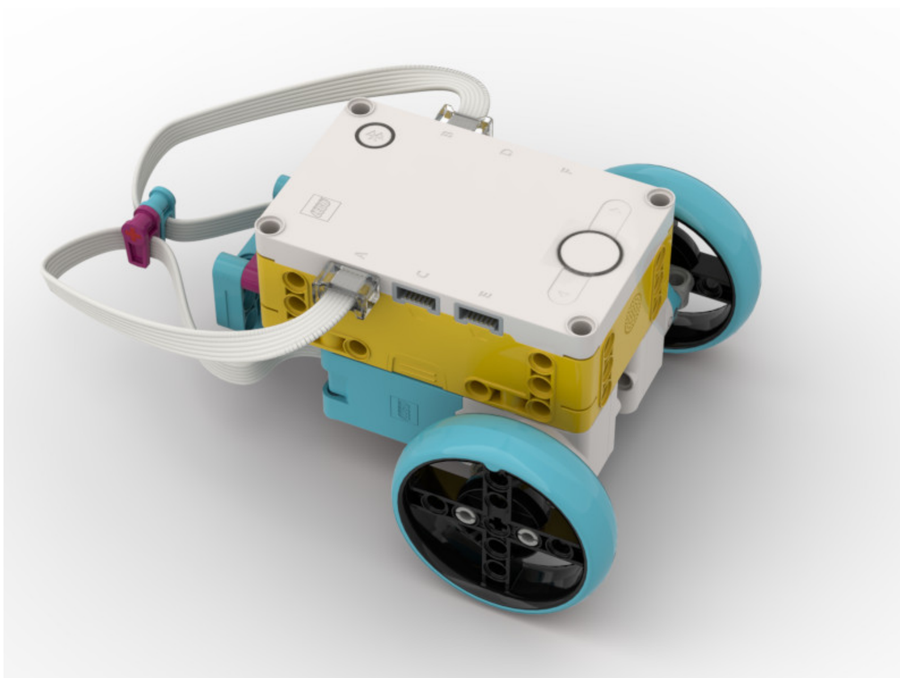
All challenges follow a similar structure:

- Scenario setup + background information. Teachers are free to develop each scenario further as they see fit.
- Equipment list. Aside from the standard SPIKE robotics kit, all other required resources are easily sourced within a school environment.
- Teacher notes are provided on common issues that may arise with each challenge and how they are best dealt with.
- Programming examples in the SPIKE software environment.
- Student worksheets to fill out (photocopy / print permission is provided).
- Extension activities.

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Chapter 2: SPYKEE Basics

Overview: Build a robot that is capable of driving around an obstacle course.



Project: THE SPACE AGENCY is in the market for a new planetary rover to explore the recently discover planet Tobor-3. You are required to construct and test a robot that can follow a set of commands to explore the planet's surface. Before the robot is deployed, it must be extensively tested to ensure it will perform as expected. You can't fly a technician to Tobor-3 to reboot the robot!

Equipment required

- 1 SPIKE robot kit per group
- 1 computer per group
- Masking tape and Tape measure

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Teachers' Notes

This section will cover the following topics amongst others

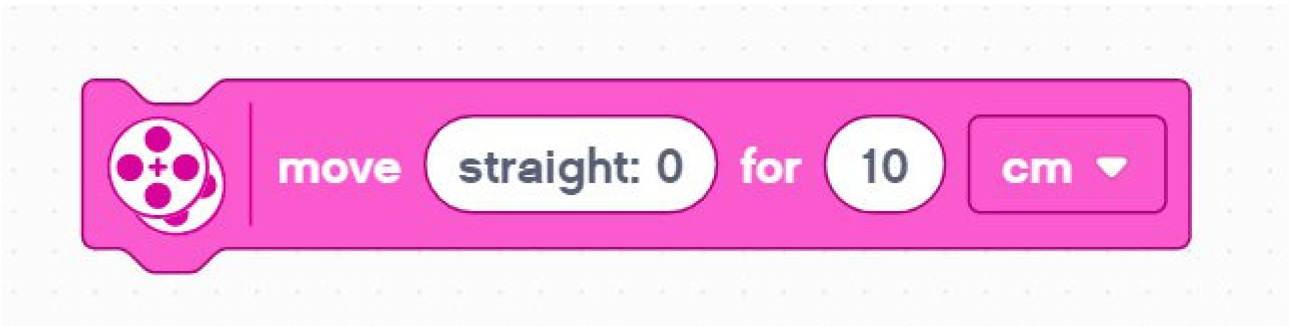
- Basic numeracy
- Decimal and fractional numbers
- Relationship between diameter and circumference
- Conversion between millimetres and inches

Get the students to build SPYKEE robot presented in Building Instructions.

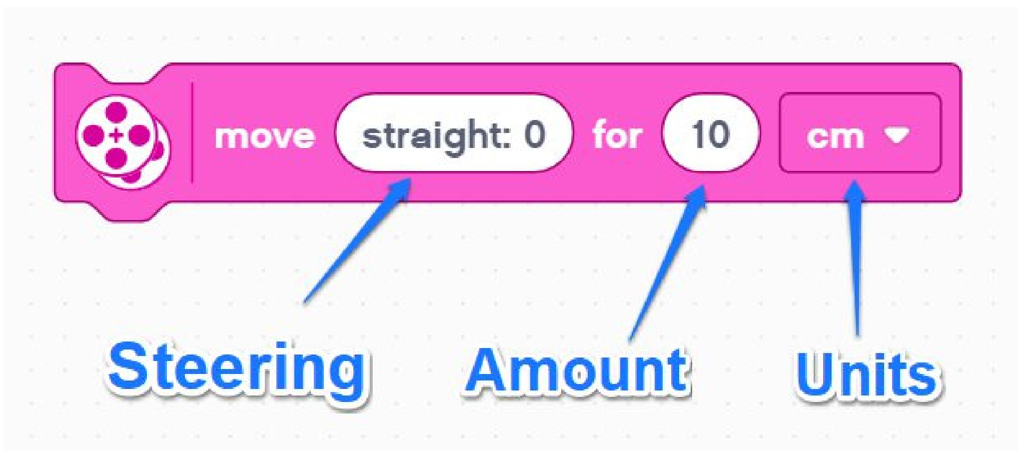
Photocopy and hand out Student Worksheet – SPYKEE Basics. This worksheet gives the students a range of different activities to follow that progressively increase in difficulty. To make our robot move, we need to send instructions to the motors which in turn drive the wheels. The SPYKEE design is often referred to as a wheelchair configuration, as it has a Left and Right motor that allows the robot to drive forwards, backwards and make turns.

SPIKE Software Specific

To perform the programming, we will need to know about the Move Blocks, located in the Movement Palette (Pink).



The Move Block has three parts to it as shown below. Steering, Amount and Units



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

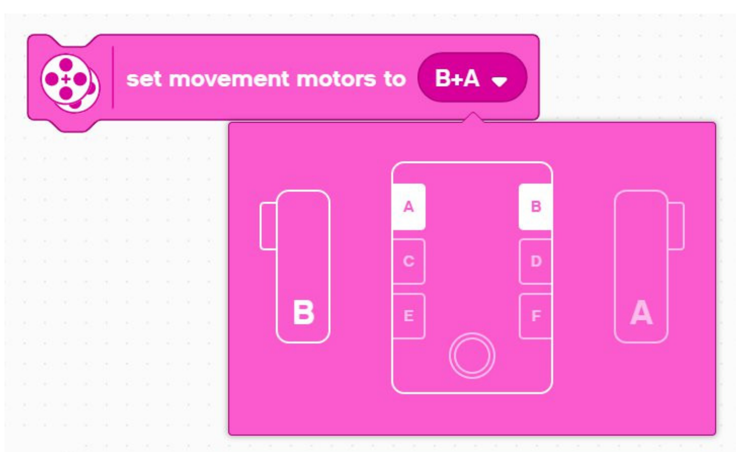
Steering: Click and drag the wheel to set how sharply the robot will turn. A steering of '0' results in the robot driving in a straight line. 100 means a sharp turn right and -100 equates to a sharp turn left. Numbers in between will allow to robot to perform gentle to sharp turns.

Amount: This section controls how far the robot will travel along the given steering curve.

Units: cm / inches/ rotations / degrees / seconds: This specifies how much the robot will travel, in either centimetres, inches, rotations (of the wheel), degrees (of the wheel) or seconds.

Before we can run our first program, we need to let the software know which ports our motors are connected to.

For our robot, we have the left motor connected to the 'B' port and the right motor connected to the 'A' port. The Set Movement Motors block is used to let the software know of this setup. Notice how this is set up as B+A, ***not*** A+B



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Why do we have Robots?

There are many reasons that robots are used in society, each one filling a need. This question may also be posed as:

"What advantages are achieved by having robots in certain situations?"

Robots are generally built to serve for what is commonly known as the 3 D's; Dull, Dirty and Dangerous.

In an industrial setting, the use of robots allows repetitive tasks to be performed accurately time after time. Robots can generally perform simple tasks far quicker than humans can. This leads to increased productivity and better quality control of goods. Some types of robots, particularly those that need to pick up and put down fragile items, are so accurate that they can stop within a human hairs width of the objects they need to manipulate. Medical robots are reaping the benefits of such accuracy, allowing doctors to perform surgery on patients who are in another city or on the other side of the world.

Exploratory robots and military robots are designed to keep people away from harmful situations. Robot operators can drive a robot into an unsafe area and use the sensors and cameras on board to gather information. This is particularly useful for search and rescue missions in disaster areas, where the environment may be unsafe for humans to go looking for survivors.

Entertainment robots are another category and provide a lot of fun and interest for people. They can be typically found on TV, highlighting the fun things that robots can do. The range of sophistication goes from the very complex humanoids such as ASIMO and NAO, to the toys like RoboSapien and the LEGO® MINDSTORMS system. Household robots such as the vacuuming Roomba was one of the first robots to be marketed as a domestic robot with later versions that have been developed to mop floors and clean out gutters. The dream of a robotic butler to pick up our clothes and do our chores is not far away.

Name different types of robots?

There are a variety of different categories for robots, including but not limited to:

- Entertainment (ASIMO, NAO, AiBO, animatronics, RoboSapien, LEGO®)
- Domestic (Roomba, automatic lawn mowers)
- Movies (C3PO, R2D2, Wall-E, Johnny 5)
- Industrial (welding, Pick and Place, factory automation)
- Medical (remote surgery, minimally invasive surgery)
- Exploratory (Mars rovers, deep sea ROV's, unassisted aerial vehicles)
- Military (PackBot, bomb disposal, search and rescue)

What are the main components of a robot?

Robots can be broken down into three distinct components; Sensors, Computation and Actuators.

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Sensors are used to 'feel' the surrounding environment. The robot uses these sensors to take in information about where it is and what it is doing. Different sensors can be used to sense different conditions including light and dark, temperature, bump sensors, ultrasonic, infrared... the list goes on and on. Think about what senses a human has, and how a robot replicates them. Sensors are classed as inputs, that is, they take information about the world and input it into the robot's brain.

The Computation component consists of an onboard computer that the robot uses to process the information coming from its sensors. This can be as small as a few integrated circuits right through to a full personal computer. The level of complexity of the required tasks will dictate the amount of computational ability needed by the robot.

The last distinct component of a robot are its Actuators. Actuators are a fancy way of saying 'parts that do things'. These may be motors in the wheels, or engines that make the arms go back and forth. It could also be hydraulic pistons or pneumatic cylinders. Actuators are a form of outputs, along with lights and speakers. The robot Computation tells these outputs to do different tasks.

The sensors provide the information to the computers, which in turn tell the motors what to do.

Path of information flow in a robot

Where did the term 'Robot' come from?

While the idea of artificial beings have been around for many years, the term 'robot' was first coined by Czech writer Karel Čapek in his play R.U.R. (Rossum's Universal Robots) in 1920. The word is derived from the Czech 'robota', which translates as 'forced work', 'slave' or 'servitude'. Čapek credits his brother Josef as the true inventor of the word.

Robots have enjoyed the majority of their exposure through movies and science fiction writings, such Star Wars and the Asimov series of 'Robot' books.

Robots in their presently accepted state were first developed in the 1950's, with George Devol's Unimate robot, capable of lifting hot pieces of metal from a die casting machine and stacking them. The first Unimate was sold to a General Motors assembly plant in New Jersey.

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Theory

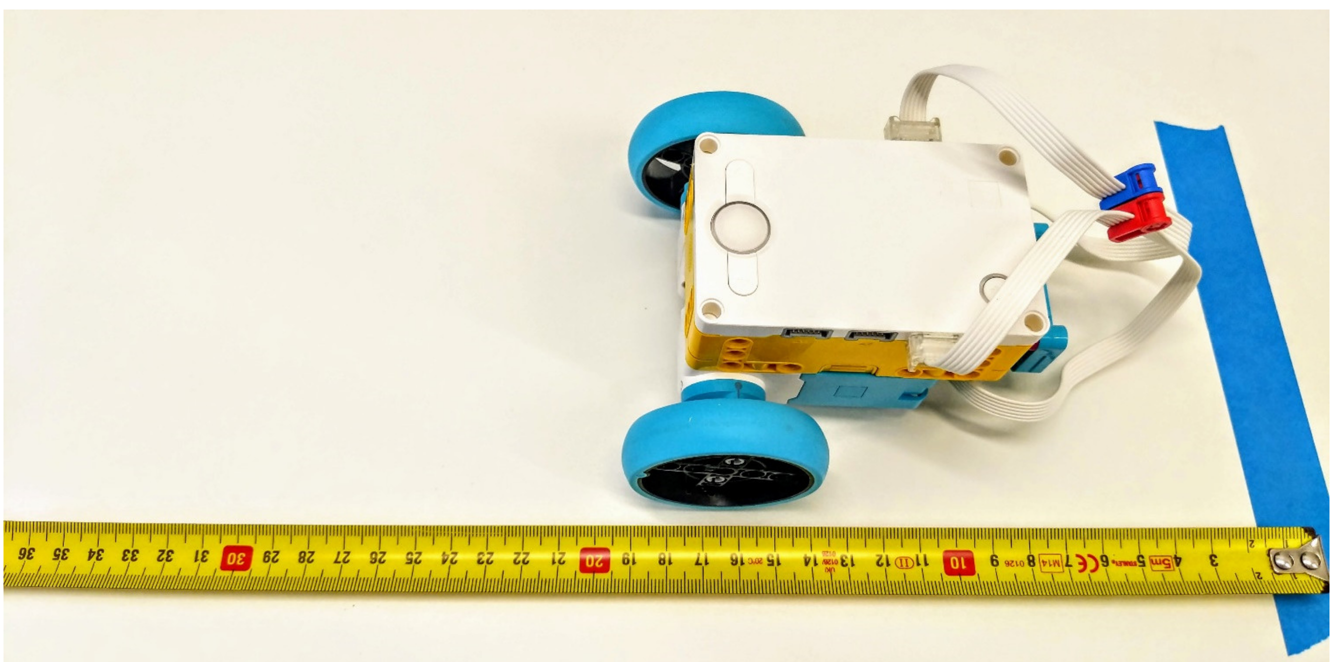
This activity will look at the effect that changing the time of travel of the robot has on the distance it moves. It will become evident that the longer a robot travels the further it travels, but can the relationship between time and distance be predicted?

Students will program their robot to travel for 1 second at a specific power level. The same experiment is run again this time for 1.5 seconds at the same power level. Students should take as many measurements as time allows with a wide variety of times. Encourage the students to take multiple runs and take the average of all their data to reduce the impact of experimental error. Keep increasing the length of time the robot is travelling and record the distance.

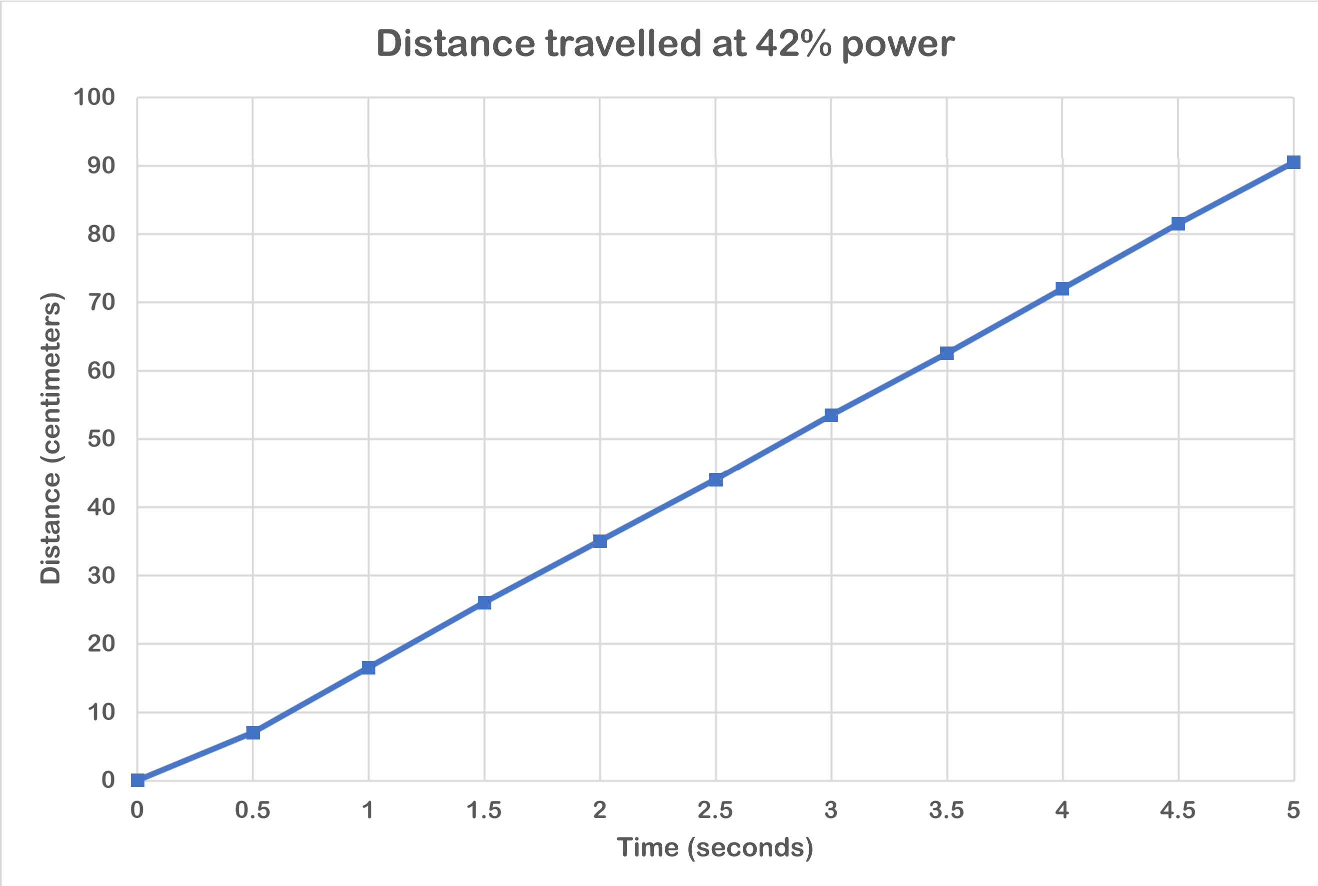
By plotting the distance travelled (vertical axis) against the time taken (horizontal axis), students build up a graph of their data. Students should find that there is a linear (straight line) relationship between the time programmed and the distance travelled. The slope of this line is the velocity of the robot (distance/time).

A random power level between 20% and 100% is assigned to each group, ensuring different results for each group. They cannot copy other groups' data as it would be inaccurate for their robot.

This data was taken for the standard SPYKEE running with a full battery. Lower battery levels will change the speed of the robot so ensure that all data is gathered in one session, using the same robot each time.



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com



Look out for...

To set up, we will need a starting line and a tape measure. Encourage the students to work out what materials they will need and gently push them in the right direction if they miss out on anything.

If this same experiment is run on carpet, students can expect to see a decrease in the distance of the robot due to the additional friction between the castor wheel and the carpet surface. Encourage this line of thought and get the students to run the same experiment with the same power level on multiple surfaces.

Be wary of very high power levels. Depending on the surface the robot is travelling on, sometimes the wheels can 'slip' as they start up, giving a slightly shorter distance than should be measured.

The Challenge!

To test that the graph is correct, place an object a random distance away from the starting line. Students will then need to read off their graph through either interpolation (reading within the graph) or extrapolation (reading beyond the graph) to determine the amount of time required to reach the marker. As an added constraint, require the robots to drive up to the marker but not knock it over.

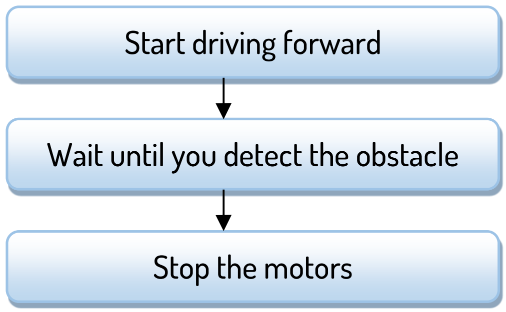
Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Teachers' Notes

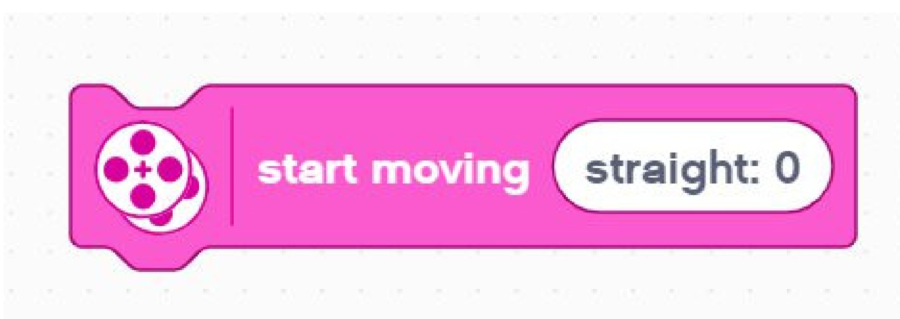
Print and handout Student Worksheet – *Help, I'm Stuck*.

The first step of this challenge is to detect an obstacle and then stop. In this case, we do not want to tell the robot's wheels to move a set distance. Moving, for example, 50 cm forward will not help us if the obstacle is 100 cm away. It would be disastrous if the object was only 20 cm away!

It helps to use a flowchart to plan what the robot will do.



This approach will still use a pink Movement Block, but in this case we will chose the **Start Moving** Block. This will turn the motors on, making the robot drive forward and immediately proceed to the next programming block. The motors will continue to drive, until another Movement Block instructs it otherwise.

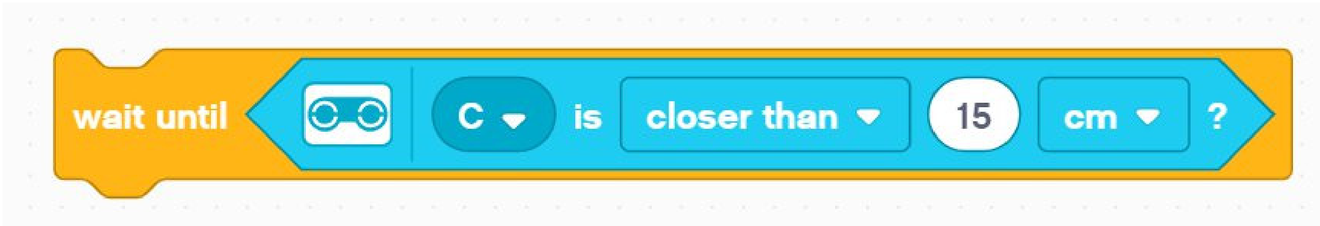


Start the robot driving. No fixed amount is specified

The Wait Block

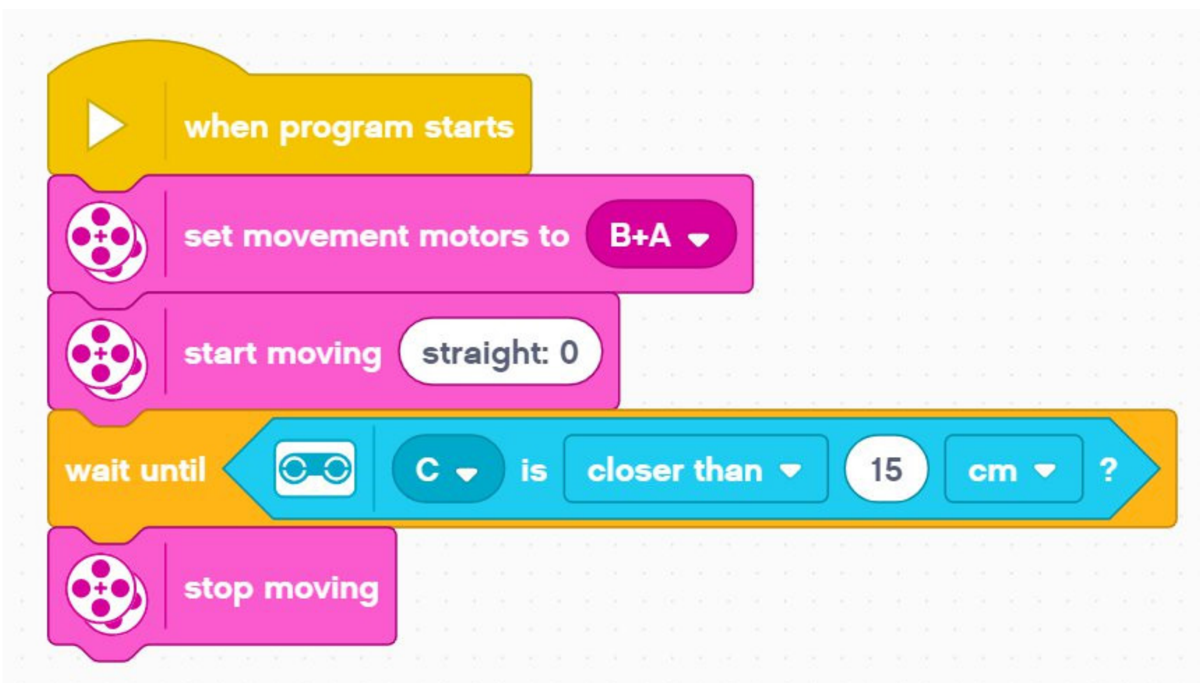
The next block will instruct the robot to wait until an obstacle has been detected. This is achieved with a **Wait Until** Block from the orange Control section. The **Wait Until** Block needs to be combined with the appropriate Sensor Block to make this program work. Notice how the blue Distance Sensor block fits within the orange Wait Until block.

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com



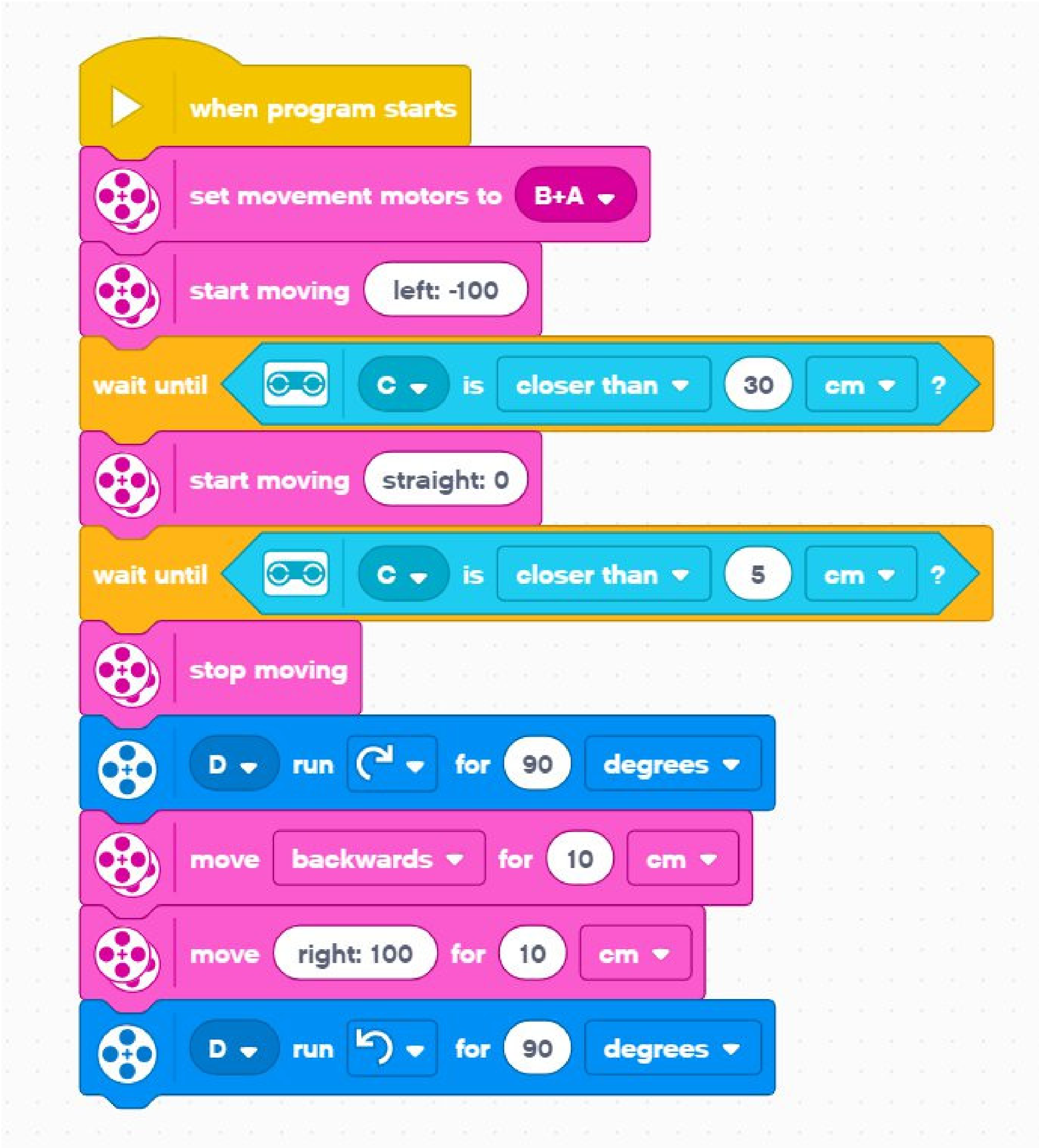
TIP: For a lot of robotics applications, we tend to avoid the 'equals' setting. It is rare, especially with sensors, that a value will ever precisely equal our requirements. Let's say our robot is moving very fast; it takes a reading of 11cm, and then by time it has a chance to take another reading, it is down to 9cm. Because the sensor never reads 'exactly' 10cm, the robot will not stop.

We're nearly finished! Now the robot starts driving and will continue driving until an object is detected less than 15cm away. If we just leave the program as is, the robot will know there is an object, but without telling the motors what to do, it will just drive into it. Our last block will tell the robot to stop moving.



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Example Program



TIP: In the above example we have closed and opened the Gripper Attachment by telling it to rotate 90 degrees. It is impossible for the arms to turn through precisely 90 degrees, as the object stops them from completely closing. The SPIKE hub will *try* to get to 90 degrees, but if it doesn't quite reach that amount, it will assume the arms are not capable of turning further and will move on to the next instruction in the program.

While it would technically do the same thing if we instructed it to turn for example 5 whole rotations, it is good to encourage students to make reasonable estimates about how far it should close.

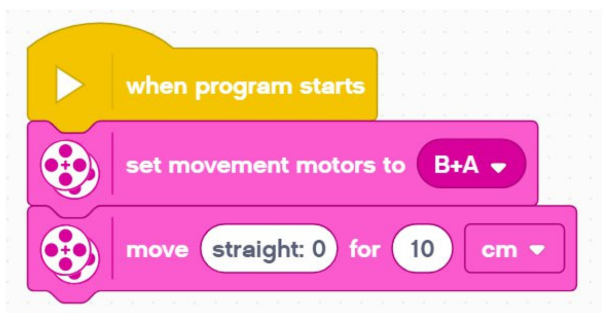
SPYKEE Basics

Group Name________________ Group Members__________________

Project: THE SPACE AGENCY is in the market for a new planetary rover to explore the recently discover planet Tobor-3. You are required to construct and test a robot that is capable of following a set of commands to explore the planet's surface. Before the robot is deployed, it must be extensively tested to ensure it will perform as expected. You can't fly a technician to Tobor-3 to reboot the robot!

Before we send our robot into space, we must first test it thoroughly here on earth. Run the following experiments and observe how your robot behaves. Do not move to the next experiment until your teacher has seen your current experiment.

Drive Forward for 50cm. Download and test the following code



Drive Forward for 2 rotations of the wheels

How far did your robot travel? ____________

Drive Forward for 2 degrees of the wheels

How far did your robot travel? ____________

Drive Forward for 2 seconds of the wheels

How far did your robot travel? ____________

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

Stay away from the Edge

Group Name__________________ Group Members____________________

Project: Another challenge the robot faces is staying safe whilst navigating on top of a large plateau. Get too close and over you'll go! THE SPACE AGENCY has asked that you prove your robot is capable of staying away from the edge of a cliff.

THE SPACE AGENCY has discovered that the Colour Sensor attachment, as well as being excellent for detecting Spiketrium, can also reliably give a reading of 'No-Colour' when it reaches the edge of the plateau. Modify your program so that the robot does not go over the edge.

There are several progressive steps we would like to make in order to solve this problem. Each program should be done individually and demonstrated to your teacher before moving on.

- Drive until the edge is detected then stop.
- Drive away from the edge and continue looking for the next edge.

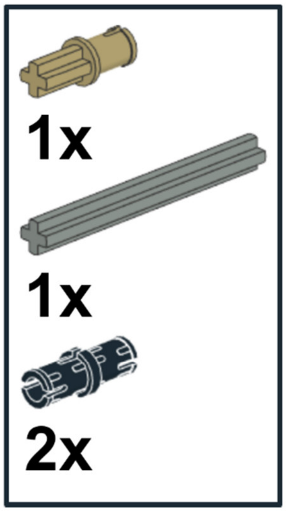
What was the most difficult part of this challenge?

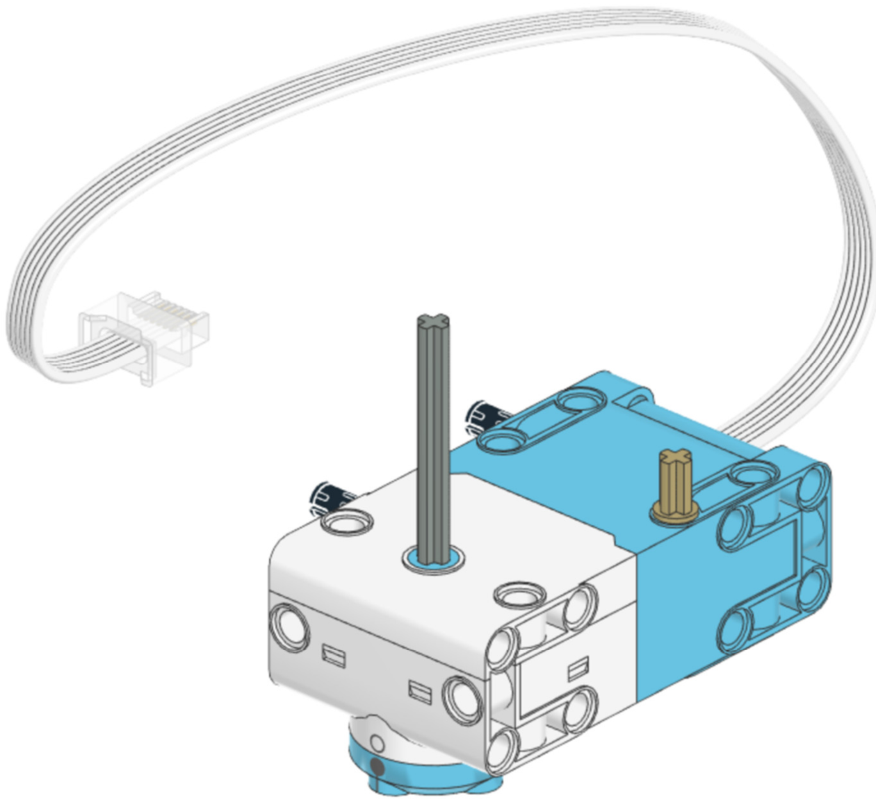
How did you go about solving it?

Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com

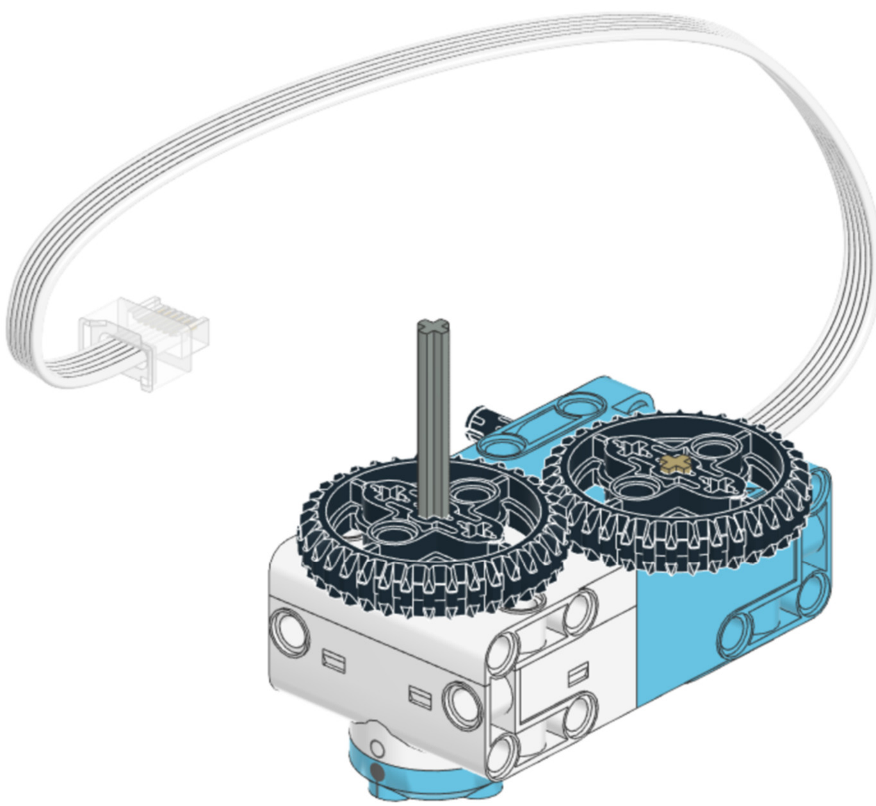
Gripper Module

1





2



Sample Pages - Classroom Activities for the Busy Teacher: SPIKE Prime - www.damienkee.com