COMPUTING KNOWLEDGE ORGRANISERS

SPRING 1

COMPUTING: PROGRAMMING

Y1

KNOWLEDGE ORGANISER

Overview

Moving a Robot

- Programming is when we make a set of instructions for computers to follow.

-Robots are one type of machine that can follow programs. Floor robots include Bee-bots and Blue-bots.

-Floor robots have buttons which help us to direct them. We can use algorithms (a set of guidelines to perform a task) to program floor robots along routes.

Robots and Floor Robots

-Robots: Robots are machines that we can program to do human jobs.

-Robots help us to do things, for example to help us clean, mow and learn!

-Robots in factories make things, and in hospitals they help make us better.

-Bee-bots: Bee-bots are a type of floor robot.
-We can programme Bee-bots to move around.

Bee-bots should only be used on the floor, and not tables etc. They can be damaged if they fall from high surfaces. (Other floor robots, e.g. Blue-bot, can also be used).

-Turning on a Bee-bot: Before we use a Beebot, we need to make sure it is charged.

To turn it on, using the switch underneath. You can tell that the Bee-bot is on because its eyes light up. Switch it back off again after you have finished using it.

Buttons and Programs

-Buttons: Bee-bots have buttons on the top. They each make the Beebot do something different (see picture).

-The arrows move the Bee-bot in different directions.

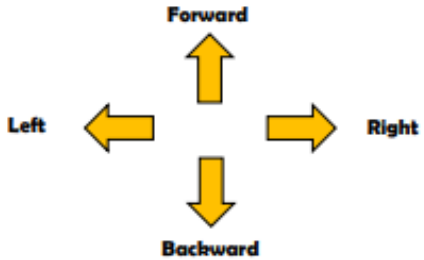
-The GO button makes the Bee-bot start its program. (on some models, it also pauses the Beebot in-program).

-Programs: A program is a series of instructions. We can program the Bee-bot by pressing the direction buttons (in order) that we want it to move in, followed by GO.

-The X button makes the Bee-bot delete the program and make a new program. Switching the Bee-bot off and on again also deletes the program.

Directions

-In order create clear routes for our Bee-bots, we need to be sure of our directions.



Make sure that you stand behind Bee-bot.

Routes and Algorithms

-A route is the course that we travel to get somewhere. We use algorithms (a set of guidelines to complete a task) to program our floor robot to take a route to where we want it to go.

-We should think carefully about how to avoid obstacles. We should also consider how many times we need to press each button to travel the correct distance.

Important Vocabulary

Programmed	Robot	Algorithm	Button	Direction	Forward	Backward	Left	Right	Route

COMPUTING: PROGRAMMING

Y2 KNOWLEDGE ORGANISER

Overview

Robot Algorithms

- Programming is when we make a set of instructions for computers to follow.

-Robots are one type of machine that can follow programs - they follow what we instruct them to do.

-We use algorithms (a set of instructions to perform a task) to help robots to do things that we want them to. Debugging can help to correct algorithms and programs.

Algorithms and Instructions

-**Algorithms**: Algorithms are precise set of instructions, that a computer can turn into a code. A floor robot has a computer inside of it.

-**Programs:** When we press the buttons of our floor robot, we are creating a program for it to follow. The program is how the algorithm is run as code on the robot.

-**Instructions:** It is important that our instructions to the floor robot are clear. If our sequence of instructions is in the wrong order, has anything missing, or has anything additional, the floor robot will end up in a different place! Plan the route to avoid obstacles and get to the right place.

Using a Floor Robot

-**Robots:** Robots are machines that we can program to do human jobs.

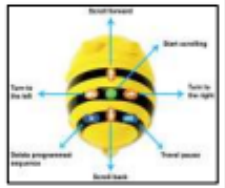
-Robots help us to do things, for example to help us clean, mow and learn!

-Robots in factories make things, and in hospitals they help make us better.

-Turning on a Bee-bot:

Before we use a Bee-bot, we need to make sure it is charged. To turn it on, using the switch underneath. You can tell that the Bee-bot is on because its eyes light up. Remember to switch it back off again after you have finished using it.

-**Buttons**: Bee-bots have buttons on the top. They each make the Beebot do something different (see picture).



-The arrows move the Bee-bot in different directions. The GO button makes the Bee-bot start its program. The X button makes the Bee-bot forget the last set of instructions.

Designing Algorithms

-We can buy or create mats for floor robots. We then need to design our algorithms so that the robot follows the given route.

-We should carefully consider the start point & end point that we want the robot to reach.

-Use symbols (e.g. arrows, crosses) to indicate the commands that will be inputted as a program.

Chunking and Debugging

-**Chunking:** With larger programs, we can break the task into chunks and create algorithms for each chunk.

-**Debugging:** Debugging is finding and fixing errors in our algorithms and programs. These errors can include:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.

-Keying errors: Typing in the wrong code.

-Logical errors: Mistakes in plan/thinking.

Important Vocabulary

Program	Robot	Algorithm	Direction	Route	Obstacle	Design	Error	Chunking	Debugging

COMPUTING: PROGRAMMING

Y3

KNOWLEDGE ORGANISER

Overview

Sequencing in Scratch

- Programming is when we make a set of instructions for computers to follow.

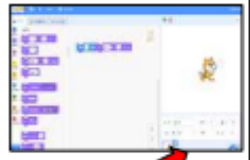
-Scratch is a program that we can use in order to code our own stories and animations.

-We use algorithms (a set of instructions to perform a task) to sequence movements, actions and sounds in order to program effective animations.

The Basics of Scratch

-What is Scratch? Scratch is a website/ app that lets us code our own stories, games and animations.

-Scratch helps us to learn how to use programming language, whilst also being creative and using problem-solving skills.



There are three main areas in Scratch:

-The Blocks Palette (on the left) contain all of the different blocks: puzzle piece commands which control the animation.

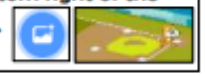
-Code Area (in the middle) is where the blocks are placed to create a program.

-Stage with Sprite (right) is where the output of the program is presented. The sprite is the character.

Adding/Removing Sprites: This can be done here, at the bottom of the stage. There are many sprites to choose from.

Attributes: There are three attributes of the sprite which we can change to make our animation: Code, Costumes, Sounds.

-Backdrops: Backdrops can be added by clicking on this icon (bottom right of the screen, below the stage).



Programming Using Blocks

-Basic Programming: Make sure that the feature of the stage that you want to program (e.g. sprite, background) is selected by clicking on it. Drag the block command that you want onto the code area. Blocks can be deleted by right-clicking on the block and selecting 'delete block.'

-Block Editing: White areas on blocks can be edited. Click on them and type in the preferred value.

-Running the Code: You can run your animation by performing the action stated in the event block (e.g. clicking the event block). If this does not work, you may need to debug your animation (find errors and fix them).

Sequencing and Algorithms

-A **sequence** is a pattern or process in which one thing follows another.

-In Scratch, blocks can stack vertically on top of one another to create sequences.

-Event blocks are used to start sequences. They are orange and have a curved shape at the top.

-Designing an **algorithm** (set of instructions for performing a task) will help you to program the sequence that you require.

Making Music

-Several sprites, each following connected sound sequences, can create music!

-In order to do this, you will need to **carefully plan your algorithm.**

-If your animation does not work correctly the first time, remember to **debug** it.

Important Vocabulary

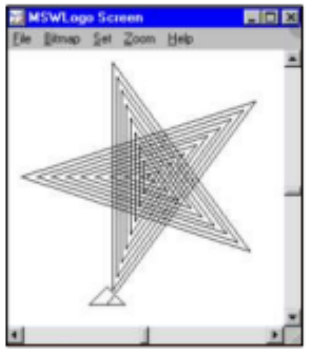
Programming	Scratch	Blocks	Commands	Code	Sprite	Stage	Costume	Backdrop	Debugging

COMPUTING: PROGRAMMING

Y4

KNOWLEDGE ORGANISER

Overview



Repetition in Shapes

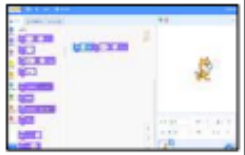
- Programming is when we make a set of instructions for computers to follow.

-Logo is a text-based program that we can use in order to create shapes and patterns.

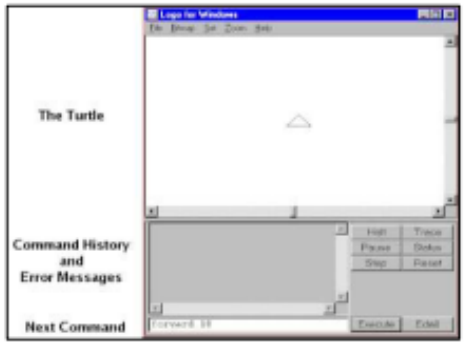
-We use algorithms (a set of instructions to perform a task) which we can plan, model and test, in order to create accurate and imaginative shapes and patterns.

The Basics of FMS Logo

-**What is FMS Logo?** Logo is a text-based programming language, where we can type commands which are then drawn on the screen.
-Logo helps us to learn how to use programming language, whilst also being creative and using problem-solving skills.



The Display:



Basic Commands:

-FD: Forwards. Always followed by a space and the number of steps, e.g. FD 50
-BK: Backwards. As above, e.g. BK 50
-LT: Left turn. Always followed by a space and then the degrees to turn, e.g. LT 90
-RT: Right turn. As above, e.g. RT 90
-CS: Clears any pen marks on your screen and gets the turtle back to the centre.
-PU: Stops turtle from leaving a pen trail.
-PD: Makes turtle leave a pen trail again.

Programming Patterns

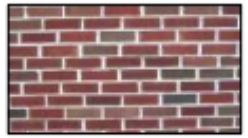
-**Patterns:** Patterns are things that repeat in a logical way. In everyday life, patterns are everywhere!

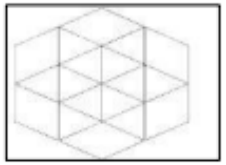
-**Patterns in Logo:** Instead of typing in the code to create each individual shape, we can save time by repeating a sequence of instructions. We use the 'repeat' function.

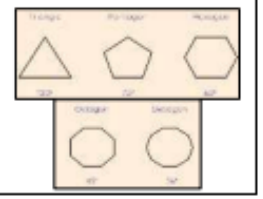
-**Repeat:** Type the command 'repeat' — this repeats commands a set number of times. The number following repeat is the number of times to repeat the code, and the code to be repeated is in square brackets, e.g. repeat 4 [FD 100 LT 90]

The above code will repeat FD 100 LT 90 four times.

-**Creating Shapes and Loops:** To make shapes, we need to know the angles of corners of different shapes (see right). Using the repeat function with shapes can help us to make spirals.







Sequencing and Algorithms

-A **sequence** is a pattern or process in which one thing follows another.

-We design **algorithms** (sets of instructions for performing a task) to help us program the sequence that we require to achieve our desired outcomes.

-**Programming** is the process of keying in the code recognized by the computer (using your algorithm).

Trialling and Debugging

-Programmers do not put their computer programs straight to work. They **trial** them first to find any errors:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.
-Keying errors: Typing in the wrong code.
-Logical errors: Mistakes in plan/thinking.

-If your algorithm does not work correctly the first time, remember to **debug** it.

Important Vocabulary

Programming	Logo	Turtle	Commands	Code	Cursor	Algorithm	Pattern	Sequence	Debugging

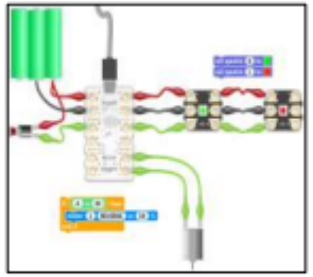
COMPUTING: PROGRAMMING

Y5

KNOWLEDGE ORGANISER

Overview

Selection in Physical Computing



- Programming is when we make and input a set of instructions for computers to follow.

-Microcontrollers are devices that can be programmed to control output devices that are connected to them.

-We use algorithms which we can plan, model, trial and debug, in order to create accurate command sequences, involving multiple output devices (e.g. LEDs and motors).

Microcontrollers, LEDs and Motors

-**Microcontrollers:** A microcontroller is a small device that can be programmed to control devices that are connected to it.

-One brand of widely used microcontroller is called a Crumble controller, which can be used to control many things, e.g. LEDs and motors.

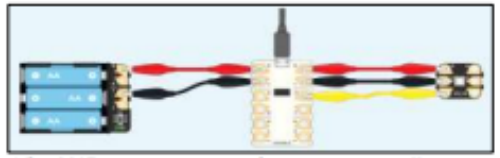
LEDs:

-LEDs are output devices that are emit light. When electricity is passed through an LED it produces light. One type of LED light, controlled by a Crumble controller, is called a Sparkle.

Motors:

-Motors are another output device. A motor can start, stop, spin forwards, spin backwards, and go at different speeds.

Creating Circuits:



-The USB port connects the microcontroller to a computer. Crocodile clips pass electricity and data through to the LED/motor.

-The + and - power pads on the Crumble should be connected with the + and - power pads on the Sparkle and battery box. The D pads on the Crumble and Sparkle should also be connected.

Programming Commands

-For programming, we should use the microcontroller software.

-Crumble uses command blocks (like Scratch).

-**Adding/Removing Commands**: To add a command block, drag it from the menu towards the program. When the grey arrow appears, the command will snap into the program. To remove a command block, drag it away from the program and back to the menu.

-**Modifying Commands:** Clicking on the colour square in the command block allows us to change the Sparkle's colour. To change the time of commands, click on the value. Delete the current value and type in the new value. Press enter after completed.

-**Count Controlled Loops:** These allow us to put programs on a loop. Count Controlled Loops are found in the 'Control' options. Drag the desired program into the Count Controlled Loop command block. 'Do until' loops allow commands to happen until a condition is met.

Sequencing and Algorithms

-A **sequence** is a pattern or process in which one thing follows another.

-We design **algorithms** (sets of instructions for performing a task) to help us program sequences involving multiple output devices (e.g. LEDs and motors).

-**Programming** is the process of keying in the code recognized by the computer into the software (using your algorithm).

Trialling and Debugging

-Programmers do not put their computer programs straight to work. They **trial** them first to find any errors:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.

-Keying errors: Typing in the wrong code.

-Logical errors: Mistakes in plan/thinking.

-If your algorithm does not work correctly the first time, remember to **debug** it.

Important Vocabulary

Programming	Circuit	Electricity	Microcontroller	Code	LED	Algorithm	Motor	Modify	Debugging

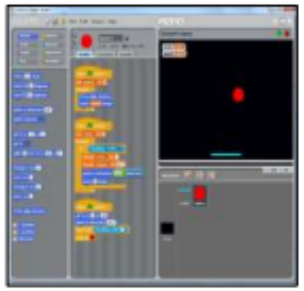
COMPUTING: PROGRAMMING

Y6

KNOWLEDGE ORGANISER

Overview

Variables in Games



- Programming is when we make and input a set of instructions for computers to follow.

-Variables are changeable elements of a program. Scratch is one app in which we can explore variables.

-We use algorithms which we can plan, model, trial and debug, in order to create accurate command sequences, that enable variables to be enacted in games.

Basic Variables

-**Variables:** A variable is something that is changeable. A variable can be set and changed throughout the running of a program.

In computer programming we use variables to store information that might change and can be used later in our program. E.g. in a game a variable could be the current score of the player; we would add 1 to the variable whenever the player gained a point.

Making Variables in Scratch – The Basics

-Select 'Variables' (dark orange circle) from the menu on the left. Either choose from the available variables or 'Make A Variable.'
-Select 'Events' (light orange circle) from the menu on the left. Choose what needs to happen for the variable to change. E.g. 'When this sprite clicked' or 'when space key pressed.'
-Select 'Variables' again from the menu on the left. Choose what will happen when the event happens, e.g. 'change score by 1' (to add a point) or 'change score by -1' to remove a point.

More Complex Variables

- Variables should always have a value and an appropriate name.

-**Adding Callouts:** Select 'Looks' from the menu on the left. Add it to the variable program. Edit the text to change the callout.

-**Adding Motion:** Many games require sprites to change position. This is achieved using the 'Motion' commands. Select 'Motion' from the menu on the left. Choose from the available motion commands.

-**Adding Motion:** Many games require sprites to change position. This is achieved using the 'Motion' commands. Select 'Motion' from the menu on the left. Choose from the available motion commands

-**Adding Comments:** Comments are a good way of showing that you understand what your code is doing. Right click on the block that you want to comment on, and add in your comment.

Sequencing and Algorithms

-A **sequence** is a pattern or process in which one thing follows another.

-We design **algorithms** (sets of instructions for performing a task) to help us program sequences involving multiple output devices (e.g. LEDs and motors).

-**Programming** is the process of keying in the code recognized by the computer into the software (using your algorithm).

Trialling and Debugging

-Programmers do not put their computer programs straight to work. They **trial** them first to find any errors:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.
-Keying errors: Typing in the wrong code.
-Logical errors: Mistakes in plan/thinking.

-If your algorithm does not work correctly the first time, remember to **debug** it.

Important Vocabulary

Programming	Variable	Scratch	Events	Code	LED	Algorithm	Motor	Modify	Debugging