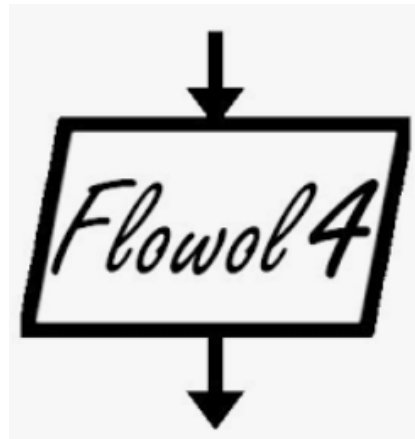
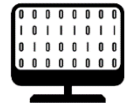




{ Spring }

Computer Science



**ICT - Year 7**

Topic: Scratch & Flowol

Name: \_\_\_\_\_




**KNOWLEDGE ORGANISER**  
**BIG IDEA: Computer Science**  
**TOPIC: Programming using Scratch and Flowo**

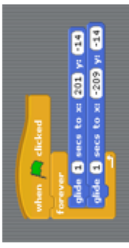
Key Word	Definition
Variable	A 'storage box' used in programming to record data
Sprite	A programmable object in Scratch
Script	A block of programming code in Scratch
Broadcast	A script used to activate other scripts during a project without needing a user prompt
Stage	The playable area / the screen
Background	The image which is like the wallpaper of the stage
Annotation	Labelling the features of your game / level
Success Criteria	What your game needs to have for it to be a success
Objectives	The aim / purpose of the game (or level)
Flowcharts	Graphical and verbal illustrations of algorithms. Useful in describing any step-by-step procedure and making it easier to create programming code
Pseudo code	Flow chart in a text form. More like a computer program, but with human language therefore it can be easily translated into computer languages.
Algorithms	Sequence of step-by-step instructions that will produce a solution to a problem – for this term we will not be playing around with Algorithm exercises

- Understand the history of computer games
- First computer game was called "Spacewar" and was released in 1971
  - Core computer games – in-depth expensive games e.g COD
  - Casual computer games – Simple/Basic e.g Tetris and Space invaders
  - Serious computer games – Complex game, like real life e.g Flight simulator
  - Educational computer games – aimed at educating the user e.g Typing

Understand how to make a scratch Character move using the direction keys on the keyboard



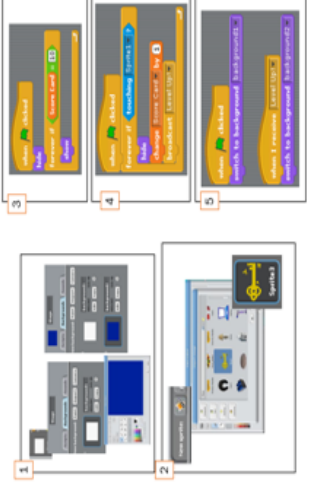
Understand how to make a sprite move automatically



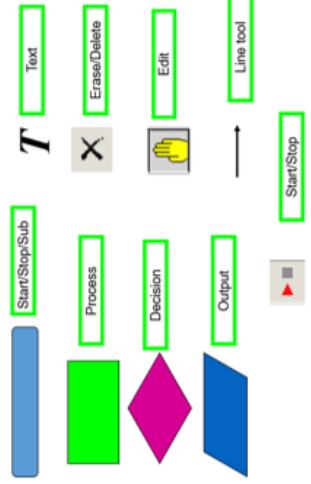
Programming a scoring item



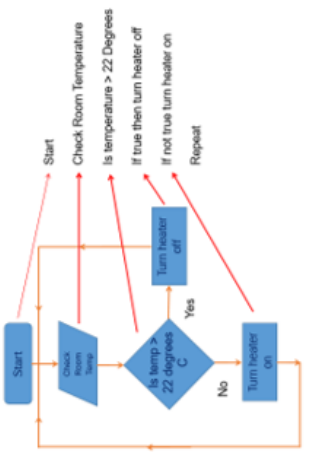
Programming levels



Be able to identify flowchart symbols



Understand a basic flowchart






## Lesson 1: Introduction to Scratch

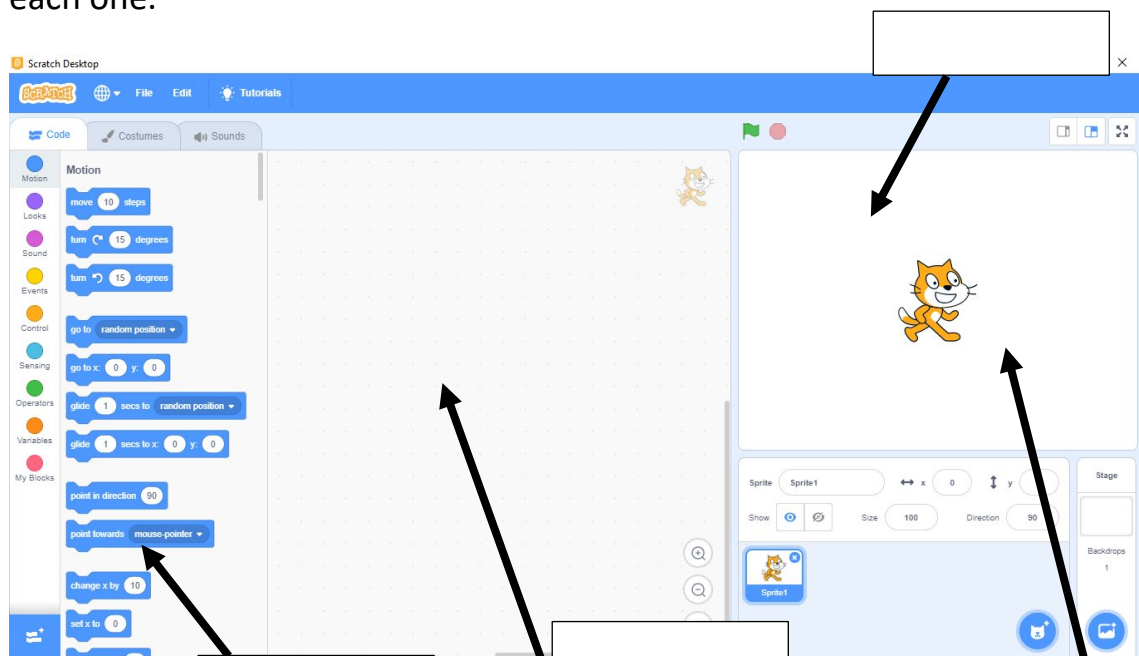
**Objective:** Understand the layout of Scratch and create basic Scratch projects

### **DO NOW:** Project Setup

- Logon to the computers and open Scratch 3.0.
- To do this, open the "4. Subject Software" folder on your desktop, then ICT and the file is in there.
- In your Year 7 ICT folder, create a Computer Science folder

### **Task 1:** Scratch layout

Identify what each of the areas of Scratch are and give a brief explanation of each one.



Key term	Explanation
Sprite	
Stage	
Control Panel	
Script area	



## Task 2: Getting to know Scratch

Get your cat sprite to:

1. Say "hello" for 2 seconds
2. Then change color by 25
3. Then think "Hmm.." for 4 seconds
4. Then change color by 75
5. Wait 3 seconds
6. Then change size by 200

Scratch Desktop

Scratch File Edit Tutorials

Code Costumes Sounds

Motion

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point forwards mouse-pointer
- change x by 10
- set x to 0

Write down the script in this box!



Get your cat sprite to:

1. Run a little way using the right arrow key
2. Turn left by 90 degrees
3. Switch the costume to "costume2"
4. Play the meow sound once
5. Spin the cat around in a circle forever

Scratch Desktop

Code Costumes Sounds

Motion

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer
- change x by 10
- set x to 0

Write down the script in this box!

### **Task 3:** Character movement

- Delete the cat and add two new characters of your own choice.
- You need to program one character to move according to the keys being pressed
- The second character needs to move around the screen automatically.

**Extension:** Develop your project by creating your own backdrop and see if you can get one character to say something if it hits the other character.



## Lesson 2: Movement and Repetition

Objective: Understand the purpose of iteration and use it to draw shapes in Scratch

**DO NOW:** For each of the Scratch commands, match them to the category you will find them in.

	<input type="radio"/> Motion
	<input type="radio"/> Looks
	<input type="radio"/> Sound
	<input type="radio"/> Events
	<input type="radio"/> Control
	<input type="radio"/> Sensing
	<input type="radio"/> Operators
	<input type="radio"/> Variables

### Task 1: Drawing shapes

Begin by importing the pen tool. Here is the code to begin drawing a square.

Your job is to finish drawing the shape. Adapt the code to draw the following shapes:

- Hexagon
- Triangle

Answer the following questions:

Write your code for a **hexagon** below:



<p>How many lines of code did it take to draw a square?</p> <hr/>	
<p>How many lines of code did it take to draw a hexagon?</p> <hr/>	
<p>How many lines of code would it take to draw a 50 sided shape?</p> <hr/>	

**Task 2:** Drawing shapes using loops

Adapt your code to use a repeat loop. Draw the following shapes:

- Hexagon
- Octagon
- Decagon

Write a definition of the word **iteration** and explain why programmers use it

**Task 2 Extension:** Using variables

You can make things more efficient by using **variables** you will learn more about this next lesson. Try following the instructions. Change the shape you draw by setting the sides to different numbers.

The image shows four numbered screenshots from the Scratch environment:

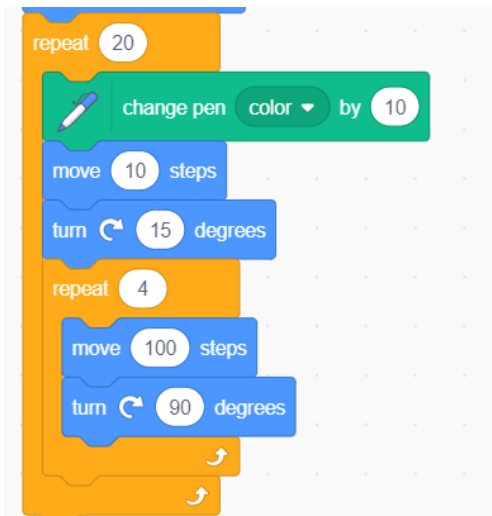
- 1:** The Scratch interface with the 'Variables' category selected in the left sidebar. The 'Make a Variable' block is circled in red.
- 2:** A 'New Variable' dialog box with 'sides' entered in the 'New variable name' field. The 'For all sprites' radio button is selected.
- 3:** A 'New Variable' dialog box with 'sides' entered in the 'New variable name' field. The 'For all sprites' radio button is selected.
- 4:** A Scratch script starting with 'when green flag clicked', followed by 'erase all', 'pen down', 'set pen color to blue', 'point in direction 90', 'set sides to 4', a 'repeat' loop containing 'move 100 steps' and 'turn 360 / sides degrees'.



### **Task 3:** Spirographs using nested repeat commands

Here is an example of a nested loop. Using this technique you can draw lots of interesting shapes

Now it is your opportunity to experiment. Use the nested repeat technique to draw different types of Spirographs



### **Plenary:** Repeat Repeat Repeat

Think about where else repetition happens in games or real life  
Give 3 examples of where else repeat commands would be useful

1	
2	
3	

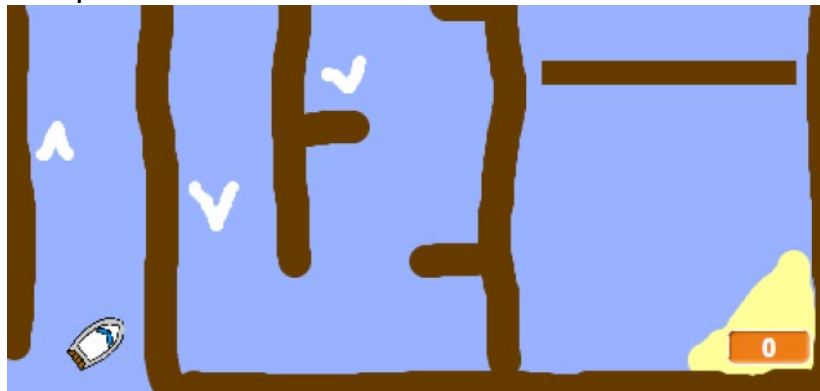


## Lesson 3: Maze Game & If statements

**Objective:** Understand the programming concept of selection and apply it to create a maze style game

**DO NOW:** This lesson you will be creating a Pacman style game. For your Pacman game, you need a simple maze in which your character can move up, down, left and right along the paths. Remember the path needs to be wide enough for your character to fit!

Here is an example maze.



Open Scratch and edit the backdrop to create your own maze.

### **Task 1:** Creating the game

You have a set of tasks to complete to finish the game. Tick them off as you go.

Make your maze	
Make/ Add your character	
Make your character move according to the arrow keys	
Make your character go back to the start <b>if</b> it touches a wall ( <i>hint</i> : put an if statement inside a forever loop)	
Put an end point at the end of your maze – make the character go back to the start	

### **Extension:** Improving your game

Well done for completing the game! Think about how you can improve it

- Can you add more obstacles to your game?
- Can you add enemies that send you to the start when you touch them?
- Fire bullets from your character?



## Task 2: Selection

Explain what selection is and why you may need to use **selection** in a program.

Write down a script from your game where you have used **selection**

Scratch Desktop

Code Costumes Sounds

Motion

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer
- change x by 10
- set x to 0

Write down the script in this box!

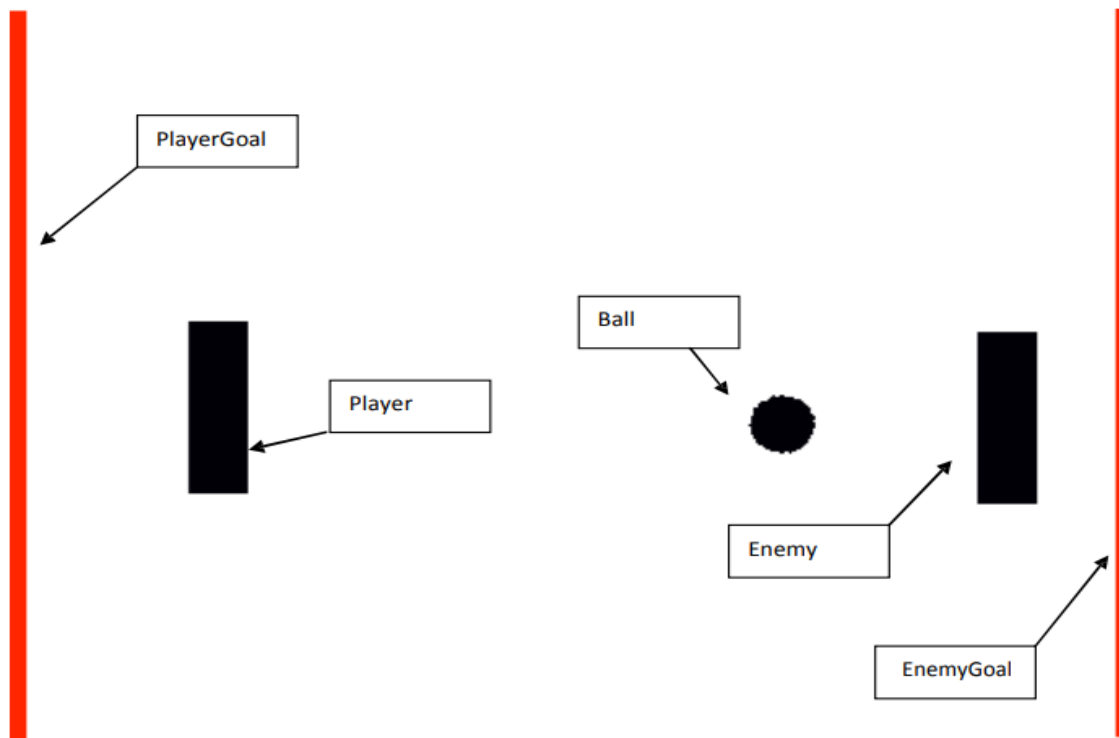


## Lesson 4: Pong & Variables

Objective: Understand the programming concept of variables and use them when developing games in Scratch

**DO NOW:** Creating the Pong sprites.

There are 5 objects you need to make in order to create Pong. Create these now in Scratch.



You can follow this tutorial if you need extra support.

<https://tinyurl.com/pongguide>

**Task 1:** Creating the game

You have a set of tasks to complete to finish the game. Tick them off as you go.

Choose a backdrop	
Make/ Add the ball	
Make the ball bounce around	
Create your paddle and the enemy paddles	
Add code to bounce the ball off the paddles	



**Task 2:** Creating a variable

- You now need to create a variable called 'score'. This should increase whenever the ball hits the players paddle/ hits the enemy goal.
- You then want to create a variable for lives. This should be set to 3 when the game starts and decrease every time it hits the player goal

**Task 3:** Variables

Explain what variables are and why you may need to use **variables** in a program.

--

Give 3 examples of other variables you might want to create in a game

1	
2	
3	

**Extension:** How much do you remember?

- Create a new project.
- Delete the cat sprite and add another of your choice.
- Make your new sprite move when you use the arrow keys.
- Add a second sprite of your choice.
- Make it move around and bounce off the screen.
- Make a variable called Score.
- If the second sprite touches the first sprite add 1 to the score and broadcast the message HideNow.
- When the first sprite receives the HideNow message, it hides for 2 seconds and then shows again.




## Lesson 5: Revision

**Objective:** Revise the different techniques you have learnt in Scratch by creating a variety of different projects

### **Challenge One:** Ball Catch Game

The ball will fall down from the air. The baby needs to try and catch it. The player can move the baby left or right using the arrow keys. When the ball is caught, 1 point is scored. If the baby doesn't catch the ball, the baby loses.

1. Set up the stage, add sprites.
2. Make the baby move left or right when arrow keys are pressed.
3. Create a variable to store the score.
4. Make sure score is set to 0 when the game starts.
5. If the ball touches the baby then add one on to score.
6. Once the baby has caught the ball, move the ball to a random position.
7. **ADVANCED:** Add a new sprite text: "Game Over"
  - a. Set its script to hide when the game starts.
  - b. Check if the ball's y position is -175 (HINT: )
  - c. If it is broadcast the message "lost" and stop the script.
  - d. For the 'Game Over Script' – if it receives the broadcast message lost, show 'Game Over'



### WHAT'S NEXT?

- Make the ball move faster
- If score is 10, change to a different ball (faster, smaller?) and a different background



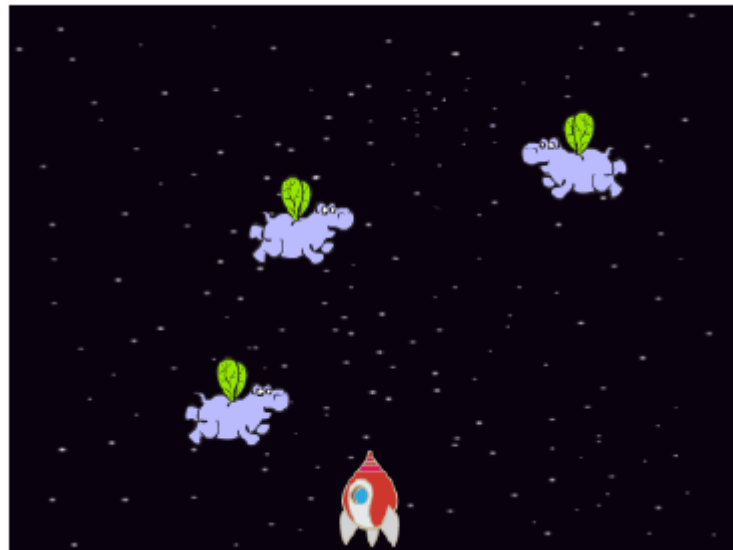
- Make the ball move in a random way (not just down). Make the baby be able to move up and down and try and chase the ball

### **Challenge 2: Clone Wars**

In this project you'll learn how to create a game in which you have to save the Earth from space monsters.

1. Setup the stage, choose your background and add sprites for the rocket and for the enemies
2. Program the rocket so it moves left and right according to the arrow keys being pressed
3. Program the enemies to move across the screen and gradually down the screen (like space invaders!)

4. Add a 'bullet' sprite. When the space key is pressed it should appear at the rocket and fly up the screen. If the bullet hits an enemy the enemy needs to disappear



5. Add a score variable – it should increase when the bullets hit the enemies
6. Add a lives variable – if an enemy makes it to the bottom of the screen it should decrease
7. If your character runs out of lives a "Game Over" screen should appear



## Lesson 6 & 7: Mini Project

**Objective:** Complete a mini project including information about all the things you have learnt this unit so far

### **Project Brief**

Over the next two weeks you will be creating your own scratch game. You will need to create a design for the game, program it and self-assess how you have done over the project.

You can create a game about anything you like but it should aim to include:

- Player movement according to the keys being pressed
- Other objects in your game (enemies, coins, balls etc) that move automatically
- Use other Scratch features such as switching between costumes/ backdrops, adding sounds
- Use variables to create a scoring/ lives system
- Any other advanced features using the techniques you have learnt



## Game Design

<u>Game Name</u>
<u>Aim / Objective of my game/ Theme of my game:</u>

<u>What parts of your game will move?</u>
<u>Explain what other objects in your game you will have (bullets, coins etc)</u>
<u>What scoring items will your game have? What levels will you have?</u>





When you have finished your game you should take a print screen on your game & code, put this on a powerpoint slide and print it off

<Stick your project in here>



## Task: Self-review

	Poor				Excellent	
	1	2	3	4	5	
Have you included player movement? <i>Are there any glitches? Is movement free or is it restricted?</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Are the characters/ sprites/ backgrounds appropriate and fit the theme of the game?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Do the other objects (enemies, coins, bullets etc.) in the game work as expected?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
How well is it coded? <i>Is progressing from level 1 to 2 smooth or are there glitches? Is game-play smooth or do objects / sprites get stuck? Does the game seem 'hugav' or does it seem 'hug free'?</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Give your game an overall score	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Are there multiple levels/ increasing difficulty	<input type="checkbox"/>	<i>No</i> ←  <i>Yes</i>				<input type="checkbox"/>
Can the players score points/ lose lives?	<input type="checkbox"/>	<i>No</i> ←  <i>Yes</i>				<input type="checkbox"/>

Explain what you think you have done well in the project:

Explain two improvements that you could make next time:



# Mini Project PRIDE

Effort

1

2

3

4

## Strength



- You've written an effective design for your game
- You have written effective player movement code
- You have written code that manages other objects (enemies, bullets, coins etc.)
- You have included a scoring/ lives system
- You have included advanced Scratch features
- You have written an effective evaluation of your project

## Target



- Add more detail to the design section of your game
- Explain what code you would use to move objects
- Write the code to move objects automatically
- Explain what a variable in programming is
- Explain what you have done well in the project
- Explain what you could do to improve the project next time

## Presentation and SPaG



- Correct any missing capital letters
- Correct your spellings
- Add in missing full stops/ apostrophes
- Pick a more suitable font/ colour so your work is easier to read
- Next time use images to support what you have written

## Student Improvement:

---

---

---

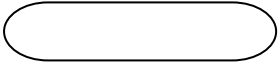

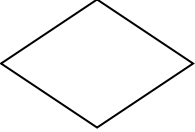
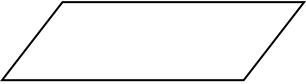
---



## Lesson 8: Introduction to Flowol

**Objective:** Understand what a flowchart is and the different symbols used and construct simple flowcharts in Flowol

**Task 1:** Complete the table below by naming each control symbol and give a brief description of what they are used for

Symbol	Name & Description
	
	
	
	



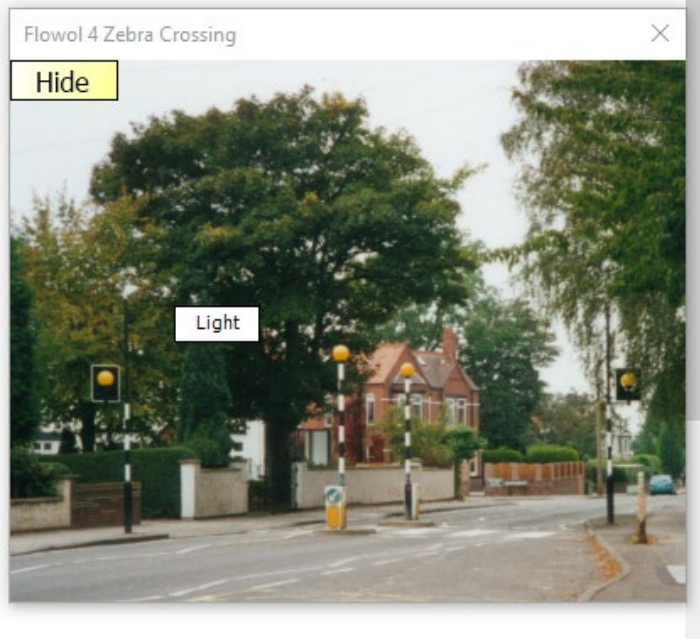
**Task 2:** Making a flowchart

You need to make a flowchart for the process of making a cup of tea. Make sure you use the correct symbols!

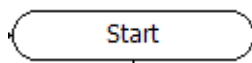


### **Task 3:** Zebra Crossing

Open Flowol 4 and choose the Zebra Crossing mimic

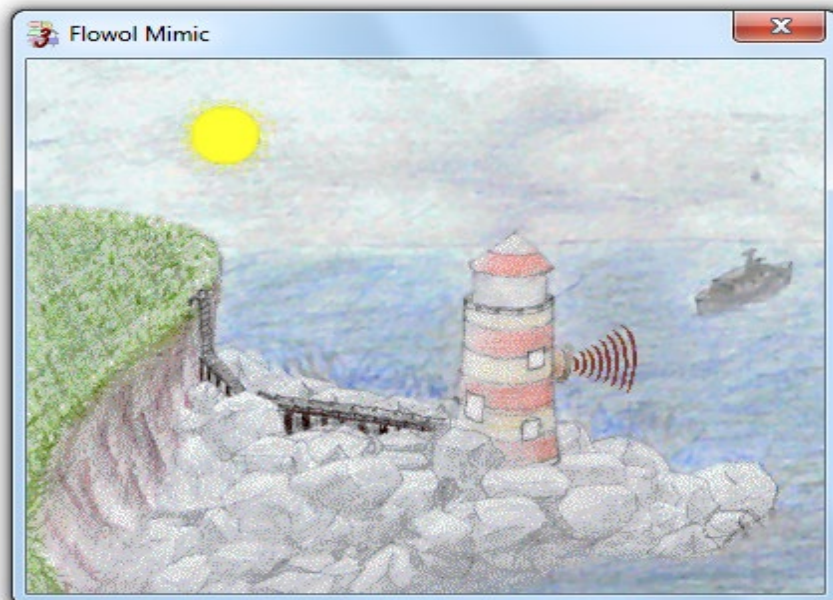


Place the symbols on the screen. Your flowchart must start with a 'start' symbol.



### **Extension:** Lighthouse

- Select the lighthouse mimic
- Use the skills you have learnt to make the lighthouse work

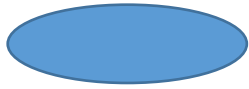




## Lesson 9: Flowol Sequencing Instructions

Objective: Use Flowol instructions in sequence to solve basic problems

**DO NOW:** Join each of the flowchart symbols to show what it represents



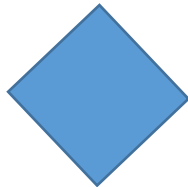
A question/decision needs to be made



The beginning or end point of an



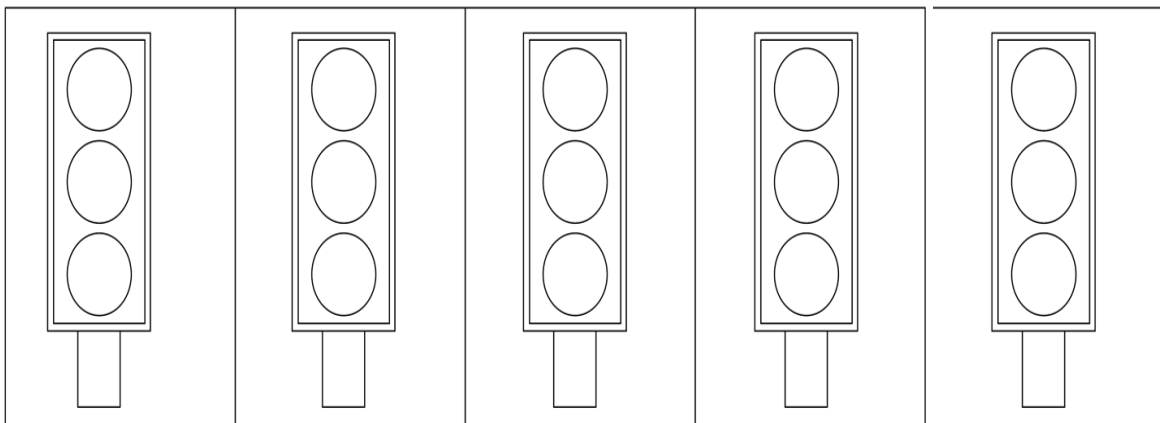
Input or output information



An activity is taking place

### **Task 1:** Traffic Lights

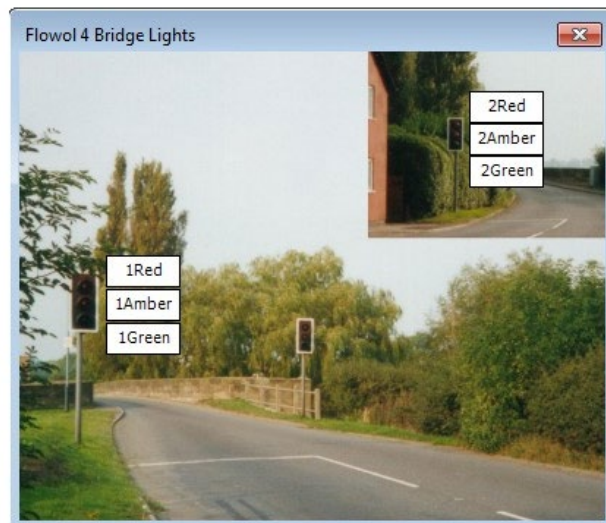
Colour the traffic lights to show the correct sequence





## Task 2: Traffic Lights

- Open the “Bridge Lights” mimic in Flowol 4.
- This mimic has two sets of traffic lights that need to play in sequence so you need to break the problem down into smaller chunks
- Create a control flowchart for the first set of lights that plays on sequence like this

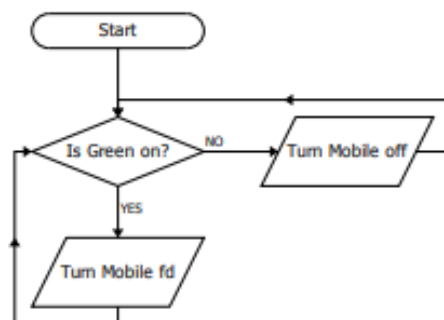
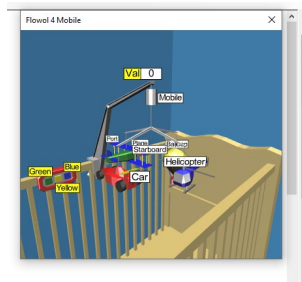


[tinyurl.com/trafficlights1](http://tinyurl.com/trafficlights1)

- Now create a second control flowchart for second set of lights  
[tinyurl.com/trafficlights2](http://tinyurl.com/trafficlights2)

## Extension: Baby Mobile

- Open the **Mobile** mimic and show its labels. Explore what it can do by clicking on the outputs and motors
- Create a flowchart to rotate the Mobile when the green button is pressed
- Construct another two flowcharts to give some forward and reverse movement to the Helicopter rotors and the Plane propeller when the Yellow and Blue inputs are used
- Now try creating flowcharts to control the rest of the Mobile





## Lesson 10: Flowol Automation

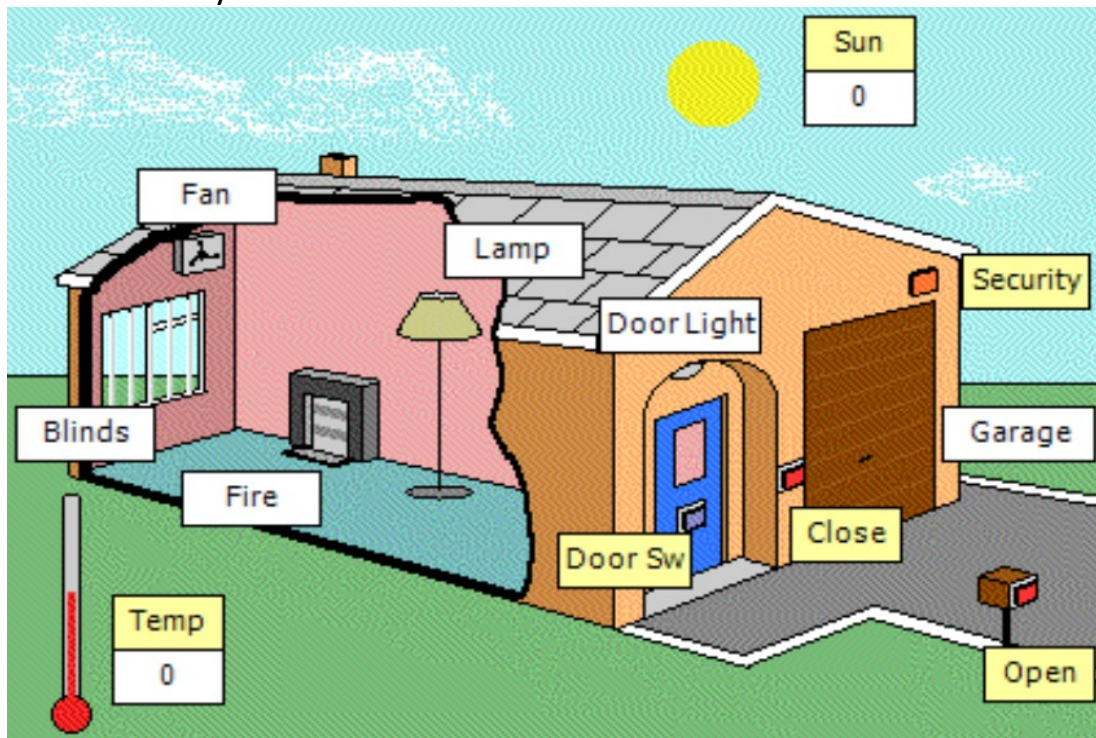
Objective: Use Flowol to understand the role of sensors and complete automation tasks

### Task 1: Sensors

Identify some areas at home or in school and explain where you would see a sensor and why is it used

### Task 2: Automatic Home

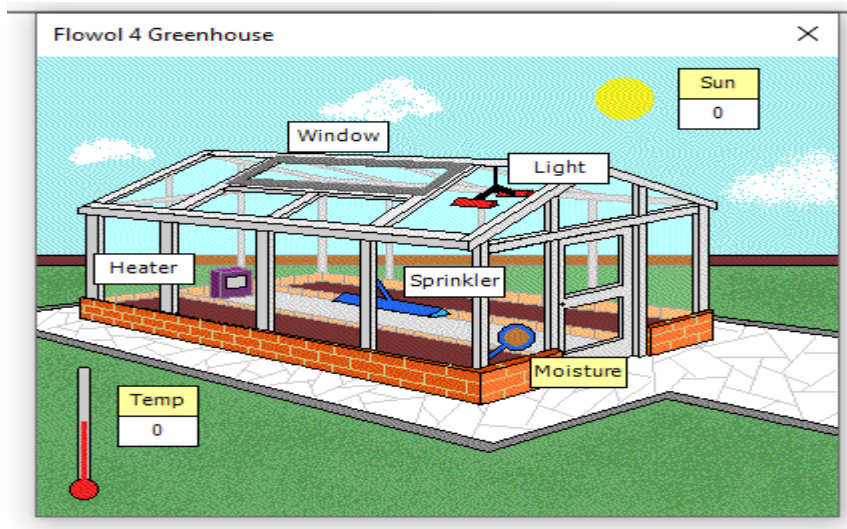
You are given a number of tasks to complete. Tick off each task once you have successfully made the flowchart





1	Turn the living room <b>fan on</b> when the temperature is <b>above 23°C</b>	
2	Turn the living room <b>fire on</b> when the temperature is <b>below 17°C</b>	
3	Turn the living room <b>lamp on</b> when the light level is <b>below 20</b>	
4	Ring the <b>doorbell</b> when the door <b>switch is pressed</b>	
5	Turn the <b>security light on</b> when the light level is <b>below 30</b> and the garage door sensor is on (detects movement)	
6	Open the <b>garage door</b> when the <b>entry button is pressed</b> and close it when the close button is pressed.	

### Task 3: Greenhouse



1	Create a flowchart to turn the <b>Heater ON</b> if the temperature is <b>less than 10</b> and turn the <b>Heater OFF</b> if the temperature is <b>greater than or equal 10</b> .	
2	Create a flowchart to turn the <b>Light ON</b> when light from the Sun is <b>less than 20</b> and turn the <b>Light OFF</b> when light from the Sun is <b>greater than 50</b> .	
3	Create a flowchart to turn the <b>Sprinkler ON</b> for <b>5 seconds</b> if the low moisture sensor is <b>ON</b> .	
4	Create a flowchart to turn the <b>Turn Window fd, Delay 0.5, Turn Window OFF</b> if the temperature is above 20	
5	Create a flowchart to turn the <b>Turn Window rev, Delay 1, Turn Window OFF</b> if the temperature is below 20	



## Lesson 11: Flowol Subroutines

Objective: Develop a control solution for a system that includes a **subroutine**

### Task 1: Lighthouse subroutine

- Load the lighthouse mimic
- Write a subroutine to make the lamp flash
- Call the subroutine from the main program

### Task 2: How a Pelican Crossing Works

Think about the things that happen at a pedestrian crossing when someone crosses the road. The green lights and red man stay on until the cross button is pressed – but what else happens? The steps that happen after the cross button is pressed are listed below with some of the words missing.

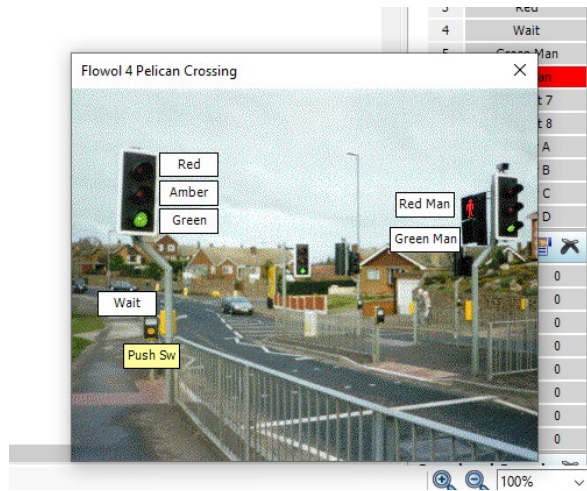
**Watch the video clips and complete the missing words below.**

- The \_\_\_\_\_ lights and \_\_\_\_\_ man are turned **ON**
- If the wait sign is turned **ON**
- There is a short delay
- The \_\_\_\_\_ lights are turned **OFF**
- The \_\_\_\_\_ lights are turned **ON**
- There is a short delay.
- The wait sign, \_\_\_\_\_ lights and \_\_\_\_\_ man are turned **OFF**
- The \_\_\_\_\_ lights and \_\_\_\_\_ man are turned **ON**
- There is a slightly longer delay
- The \_\_\_\_\_ man is turned **OFF**
- The \_\_\_\_\_ lights and \_\_\_\_\_ man are turned **ON**
- There is a short delay
- The \_\_\_\_\_ lights and \_\_\_\_\_ are turned off
- The \_\_\_\_\_ lights are turned on



**Task 3:** Pelican Crossing

- Load the “Pelican Crossing” Mimic
- Create a subroutine to control only the flashing lights
- Call the subroutine from the main program
- You then need to write the rest of the commands to complete the Pelican crossing – use your “Fill in the blanks” for support



**Plenary:** Explain what a subroutine is and give advantages for using them

A subroutine is...

**Advantages:**

- 
- 
-



## Lesson 12: End of Unit Test

**Objective: Assess our understanding of the unit by completing the end of unit test**

**Assessment Criteria:**

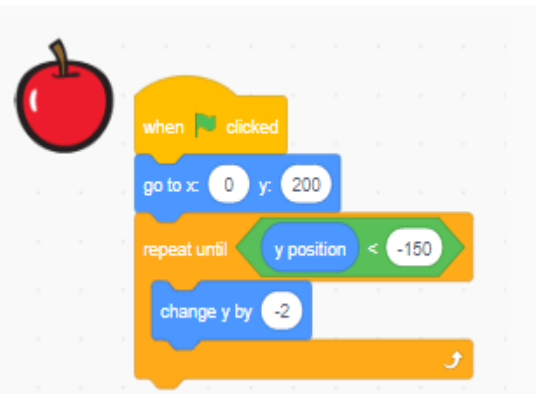
*Can add sprites, backgrounds and simple scripts to make simple games in Scratch.*

Q1) Which of these scripts will:



Demand that the user enters an input?		/3
Output some data to the screen		
Store the data that a user will enter		

Q2) What is the purpose of this script? Tick the correct answer



Launch the apple into space!		/1
Make the apple fall down		
Make the apple fly left to right		
Make the apple bounce across the screen		

Q3) What is the purpose of this script? Tick the correct answer

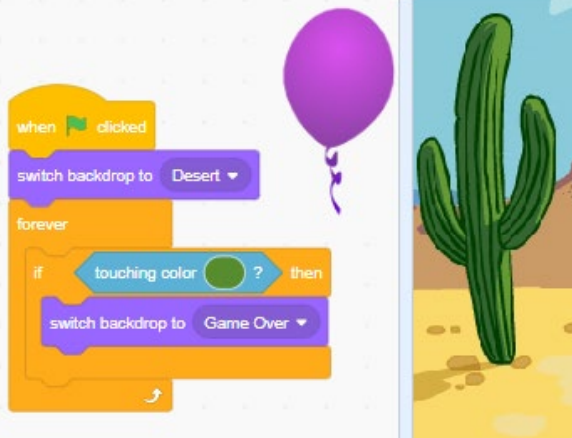


*What is the purpose of this script?*

Make the mouse bounce like a bouncing ball		/1
Make the mouse run across the screen indefinitely		
Make the mouse runaway from the cat		
Make the mouse run away and disappear off the screen		



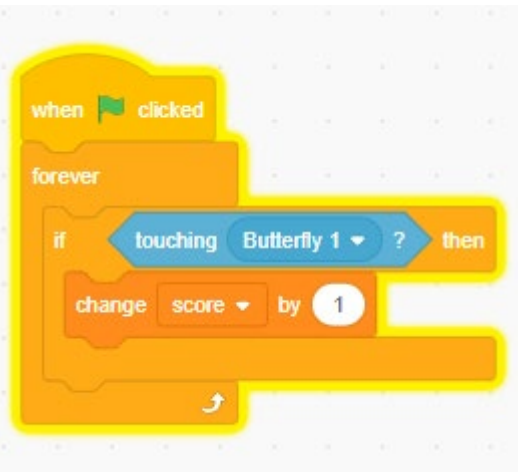
Q4) When will the game over screen appear?



When the ballon turns green	
When the user clicks on a cactus	
When the ballon touches a cactus	
When the user clicks on the ballon	

/1

Q5) Identify blocks that is/ are:



A variable	
A sprite	
Selection	
Iteration	

/4



Assessment Criteria: Can create advanced scripts in Scratch to make features such as scoring systems.

The Scratch program below uses three variables called “numOne”, “numTwo” and “numThree” to store values.

As the program runs the values stored in each variable will change. The program code and the output it generates is shown below.

Code

Output

numOne 2  
numTwo 4  
numThree 8

Note the numThree ends up with the value 8. The last line of code sets:  $\text{numThree} = \text{numOne} * \text{numTwo}$  (or  $2 * 4$ ).

Q6) State the value of each variable once the program has been run.

Working	Answers
	numOne = <input type="text"/>
	numTwo = <input type="text"/>
	numThree = <input type="text"/>

/1

Q7) State the value of each variable once the program has been run.

Working	Answers
	numOne = <input type="text"/>
	numTwo = <input type="text"/>
	numThree = <input type="text"/>

/1

Q8) Remember that a loop will cause parts of your program code to repeat. State the value of each variable once the program has been run.

Working	Answers
	numOne = <input type="text"/>

/1



Q9) An IF statement makes a decision about which code is executed next.  
State the value of each variable once the program has been run.

```

when clicked
  set numOne to 6 + 6
  set numTwo to numOne * 2
  if numTwo > 40
    set numThree to numOne
  else
    set numThree to numTwo
  
```

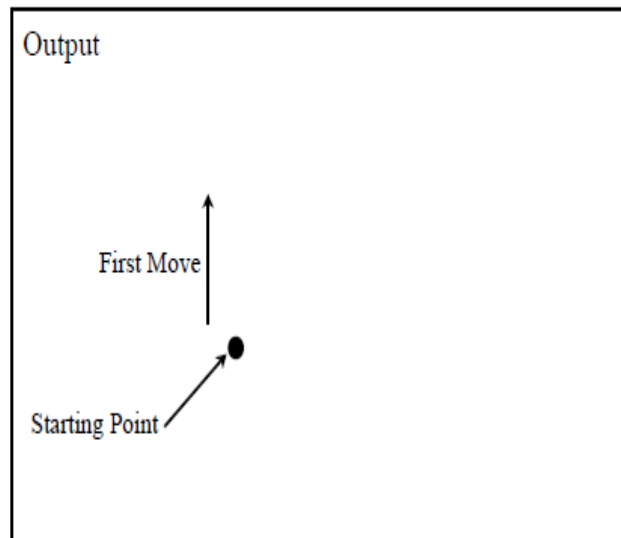
Working	Answers
<input type="text"/>	numOne = <input type="text"/>
<input type="text"/>	numTwo = <input type="text"/>
<input type="text"/>	numThree = <input type="text"/>

/1

Q10) Draw the output from the following program.

```

when clicked
  pen down
  repeat 6
    move 100 steps
    turn 60 degrees
  pen up
  
```



/3

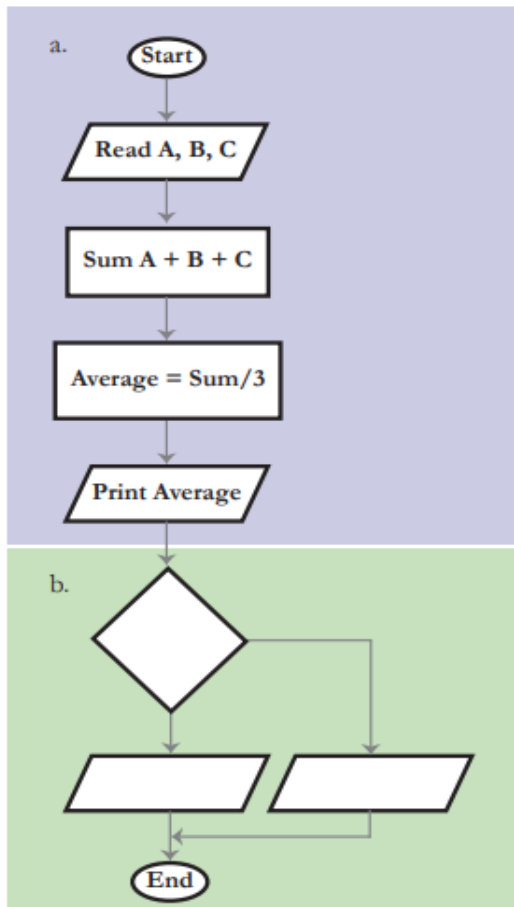
*Assessment Criteria: Can create a program in Flowol that has a simple sequence and makes simple decisions.*

Q11) Give the name and a brief description of what each flowchart symbol does


/8



Q12) A, B, C are the marks scored by a student in Science, Mathematics and English. Refer the flowchart and answer the questions.



Adam, who scores 10 marks in Science, 8 marks in mathematics and 3 marks in English.

Use the flowchart and provide the execution steps in calculating the average marks of Adam. The first step is given:

Step 1: Read: **10, 8, 3**

Step 2:

Step 3:

Step 4: Print:

A student whose average marks is above 75 is awarded a star. Fill in the empty boxes in the flowchart, choosing from the options below:

Yes

No

Award a star

Do better to get a star

Is average greater than 6

/3

Is Adam awarded a star? Yes or No?

/3

/1

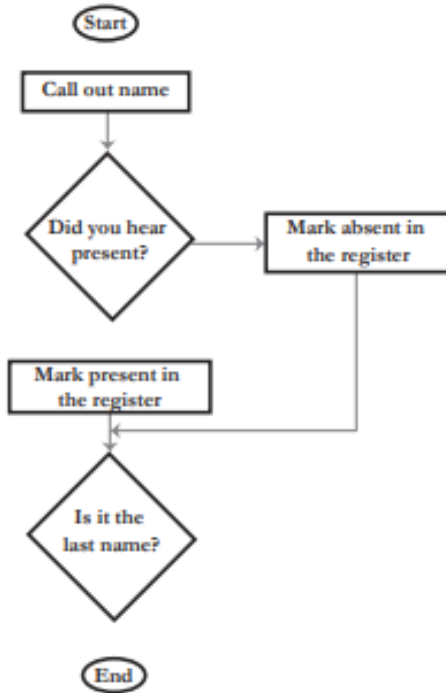


*Assessment Criteria: Can create a program in Flowol that has advanced control features such as loops and procedures.*

The following flowchart gives the steps followed while taking attendance in a class. Three arrows and a loop are missing.

Complete the flowchart as follows:

- Draw the three missing arrows and loop at the correct place.
- Label the decision arrows with yes or no.



/5

*Explain what a subroutine is and give an advantage for using them*

---

---

---

---

---

/3

/40



## Self-Reflection

Beginner	Progressing	Embedded
I am still learning this and don't feel confident about my knowledge of the topic	I am getting there but don't understand everything about the topic	I have a really good knowledge of this topic and can answer questions confidently

Assessment Criteria	Beginning	Progressing	Embedded
Can add sprites, backgrounds and simple scripts to make simple games in Scratch.			
Can create advanced scripts in Scratch to make features such as scoring systems.			
Can create a program in Flowol that has a simple sequence and makes simple decisions.			
Can create a program in Flowol that has advanced control features such as loops and procedures.			

Complete the boxes below, identify things you have done well this term and things you want to improve on. This can be specifically about the topic itself or your own working practices

Things that I have done well are:

Things I need to improve are:



# Assessment PRIDE

Effort

1

2

3

4

## Strength



- Can interpret Scratch scripts and explain what they do
- You know programming key terms such as selection, iteration variables
- Can follow programs in Scratch that use variables
- Can identify the components of flowcharts
- Can write the commands for and follow simple flowcharts
- Can use flowchart features such as looping and subroutines

## Target



- Correct your mistakes
- Give the definition for \_\_\_\_\_
- Explain what a variable is and give two examples of where they can be used
- Draw the four main flowchart symbols again and give their name
- Correct your answers to Q12. Explain where you went wrong and what you did to fix it
- Explain what a subroutine is. Give an example of where you would use them

## Presentation and SPaG

- Correct any missing capital letters
- Correct your spellings
- Add in missing full stops/ apostrophes
- Write in black pen
- Draw diagrams in pencil

## Student Improvement:

---

---

---

---