INDIAN SCHOOL AL WADI AL KABIR

CLASS IX	DEPARTMENT OF COMPUTER SCIENCE	
TOPIC :Different scratch Blocks ,Arithmetic operators, and mathematical function in scratch		Handout5

Scratch Blocks: An Overview

In this section, you'll learn about the different blocks available in Scratch, their names, and their intended usage. The goal is to define some of the terms that you'll read in the next chapters. You can come back to this sec-tion as you progress if you need to refresh your memory.

As shown in Figure 1-25, Scratch has four kinds of blocks: command blocks, function blocks, trigger blocks, and control blocks. *Command blocks* and *control blocks* (also called *stack* blocks) have bumps on the bottom and/or notches on the top. You can snap these blocks together into stacks. *Trigger blocks,* also called *hats,* have rounded tops because they are placed at the top of a stack. Trigger blocks connect events to scripts. They wait for an event—such as a key press or mouse click—and run the blocks underneath them when that event happens. For example, all scripts that start with the **when green flag clicked** block will run when the user clicks the green flag icon.
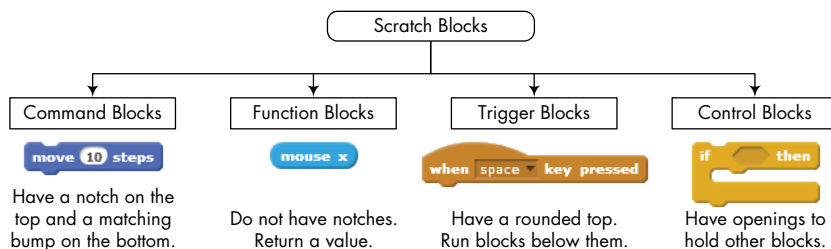

Figure 1-25: The four types of blocks available in Scratch

Function blocks (also called *reporters*) don't have notches or bumps. They can't form a layer of a script alone; instead, they're used as inputs to other blocks. The shapes of these blocks indicate the type of data they return. For example, blocks with rounded ends report numbers or strings, whereas blocks with pointed ends report whether something is true or false. This is illustrated in Figure 1-26.
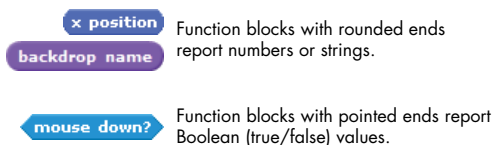

Figure 1-26: The shape of a function block indicates the type of data it returns.

Some function blocks have a checkbox next to them. If you check the box, a *monitor* appears on the Stage to display the current value of the reporter. Select a sprite and check the box on the **x position** block (in the *Motion* palette). Then drag the sprite around the Stage and watch that mon-itor. It should change as you move the sprite back and forth.

Arithmetic Operators and Functions

Now, let's take a quick look at the arithmetic operators and functions sup-ported in Scratch. If you've lost your calculator, then your worries are over! You could make your own calculator in Scratch with the blocks from the *Operators* palette, which you'll explore in this section.

Arithmetic Operators

Scratch supports the four basic arithmetic operations of addition (**+**), subtraction (**-**), multiplication (*****), and division (**/**). The blocks used to perform these operations, called *operators*, are shown in Figure 1-27. Since these blocks produce a number, you can use them as inputs to any block that accepts numbers, as demonstrated in this figure.
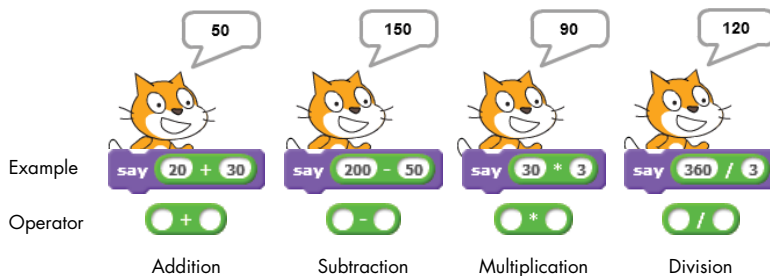

Figure 1-27: Arithmetic operators in Scratch

Scratch also supports the modulus (**mod**) operator, which returns the remainder of the division of two numbers. For example, **10 mod 3** returns 1 because the remainder of dividing 10 by 3 is 1. A common use of the modulus operator is to test the divisibility of one *integer* (whole number) by another (smaller) integer. A modulus of 0 indicates that the larger num-ber is divisible by the smaller number. Does this give you an idea for check-ing whether a number is even or odd?

Another useful operator supported by Scratch is the round operator, which rounds decimal numbers to the nearest whole number.
For example, round(3.1) = 3, round(3.5) = 4, and round(3.6) = 4.

Random Numbers

As you program more often, you'll probably need to generate random num-bers at some point, especially if you create games and simulations. Scratch provides the pick random block specifically for this purpose.

This block outputs a random number each time you use it. Its two edit-able white boxes allow you to enter a range for that number, and Scratch will only choose values between the two limits (inclusive). Table 1-1 shows some examples of using this block.

Table 1-1: Examples of Using the Pick Random Block

Example	Possible Outcome
pick random 0 to 1	{0, 1}
pick random 0 to 10	{0, 1, 2, 3, ... , 10}
pick random -2 to 2	{–2, –1, 0, 1, 2}
10 * pick random 0 to 10	{0, 10, 20, 30, ... , 100}
pick random 0 to 1.0	{0, 0.1, 0.15, 0.267, 0.3894, ... , 1.0}{0,
pick random 0 to 100 / 100	0.01, 0.12, 0.34, 0.58, ... , 1.0}

*The outputs of **pick random 0 to 1** and **pick random 0 to 1.0** are different. The first case will give you either a 1 or a 0, but the second gives a decimal value between 0 and 1. If any input to the **pick random** block contains a decimal point, the output will also be a decimal value.*

Mathematical Functions

Scratch also supports a large number of mathematical functions. The **sqrt of** block from the Operator palette contain 14 math functions that can be selected from the drop-down menu, including square root, trigonometric, logarithmic, and exponential functions. Table 1 briefly describes these functions.

Table 1: Scratch's Mathematical Functions

Function	Description
`abs ▾ of x`	Returns the absolute value of x. For example, $abs(5) = 5$, $abs(0) = 0$, and $abs(-4) = 4$. Geometrically, $abs(x)$ is the distance between x and 0 on the number line. Similarly, $abs(x - y)$ is the distance between x and y on the on the number line.
`floor ▾ of x`	Returns the largest integer that is less than or equal to x. For example, $floor(2.1) = 2$, $floor(2.9) = 2$, $floor(-2.1) = -3$.
`ceiling ▾ of x`	Returns the smallest integer that is greater than or equal to x. For example, $ceiling(2.1) = 3$, $ceiling(2.9) = 9$, $ceiling(-2.1) = -2$.
`sqrt ▾ of x`	Returns the square root of x. This is another number y such that $y^2 = x$. For example, $sqrt(16) = 4$, $sqrt(2) = 1.4142$, and $sqrt(0) = 0$. Passing a negative value for x returns NaN (short for "not a number").
`sin ▾ of x`	Returns the sine of x, where x is an angle expressed in degrees. For example, $sin(0) = 0$, $sin(30) = 0.5$, and $sin(90) = 1$.
`tan ▾ of x`	Returns the cosine of x, where x is an angle expressed in degrees. For example, $cos(0) = 0$, $cos(60) = 0.5$, and $cos(90) = 0$.
`tan ▾ of x`	Returns the tangent of x, where x is an angle expressed in degrees. For example, $tan(0) = 0$, and $tan(45) = 1$.
`asin ▾ of x`	Returns the inverse sine, or arcsine, of x. The arcsine of x is the angle whose sine is x. For example, $asin(0.5) = 30$.
`acos ▾ of x`	Returns the inverse cosine, or arccosine, of x. The arccosine of x is the angle whose cosine is x. For example, $acos(0.5) = 60$.
`atan ▾ of x`	Returns the inverse tangent, or arctan, of x. The arctan of x is the angle whose tangent is x. For example, $atan(1) = 45$.
`ln ▾ of x`	Returns the natural logarithm of x. For example, $ln(2.718) \approx 1$.
`log ▾ of x`	Returns the base-10 logarithm of x. For example, $log(1000) = 3$.
`e ^ ▾ of x`	Returns the exponential function of x. For example, $e^1 \approx 2.718$.
`10 ^ ▾ of x`	Returns 10 to the power of x. For example, $10^2 = 100$.