



# EL ÁREA DE LAS FIGURAS



## STEMJAM Teaching Guide

Developing make spaces to promote creativity  
around STEM in schools

Acronym: STEMJAM

Project no. 2016-1-ES01-KA201-025470

[www.stemjam.eu](http://www.stemjam.eu)



Co-funded by the  
Erasmus+ Programme  
of the European Union

# EL ÁREA DE LAS FIGURAS

## RESUMEN

La actividad consiste en calcular el área de unas figuras (rectángulo, paralelogramo, rombo) utilizando el robot, equipado con el sensor Sigue Líneas. Los alumnos dispondrán de unas tablas con la velocidad del robot en [cm/s], según la potencia de los motores.

## DIDACTIC OBJECTIVES

Identificar diferentes figuras geométricas y conocer el área y perímetro de cada una de ellas.

Materia STEM:            Ciencia             Tecnología             Ingeniería             Matemáticas

Nivel educativo:            12-14 años             14-16 años

## PLANTEAMIENTO DEL PROBLEMA

### Versión 1:

El mBot cuenta el área de la figura (delimitada por una cinta negra). Los alumnos utilizarán datos de la actividad de la guía “7 – Cálculo de velocidades”. La idea principal radica en utilizar la fórmula:

$$\text{espacio} = \text{velocidad} \cdot \text{tiempo} = s \cdot t$$

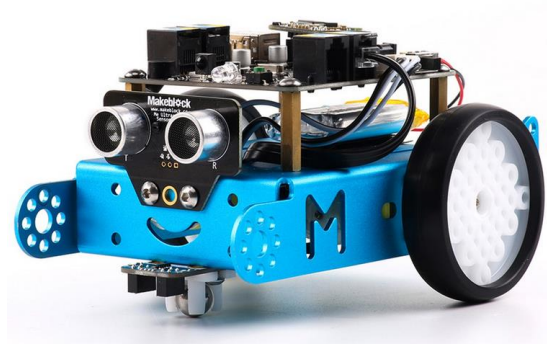
### Versión 2:

Ahora los alumnos utilizarán un sensor de Ultrasonidos para para medir la distancia. El mBot ayudará a los estudiantes a calcular las distancias, así como las áreas y los perímetros, ya que les pedirá que lo coloquen en la posición ideal para poder realizar la operación de manera independiente.



## LISTA DE MATERIALES

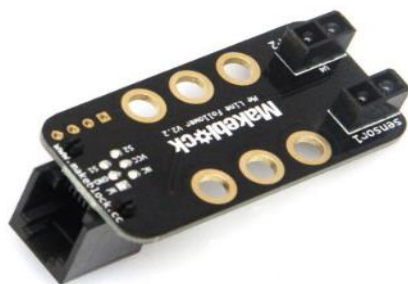
### ➤ Robot mBot



### ➤ Me Matriz LED 8 × 16:



### ➤ Me Sensor Sigue Líneas:



### ➤ Me Sensor de Ultrasonidos:



- Figuras o 3D o dibujadas sobre una cartulina, con cinta negra.
- Cartulinas blancas.

ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT1	P.MOT2
				Y	B	W	Y	B	W	Y	B	W	Bl	Y	B	W	Bl	W*	W*
Mbot Robot 2'4G			1																
Motor 1	W*		1													W*			
Motor 2	W*		1														W*		
Me Led Matrix 8x16	B	(1)	1	B															
Me Line Follower	B	(1)	1				B												
RJ25 cables			2																
Structures and beams																			
Laptops		usb	1																
Attrezzo (not essential)																			



## DESCRIPCIÓN DE LA ACTIVIDAD

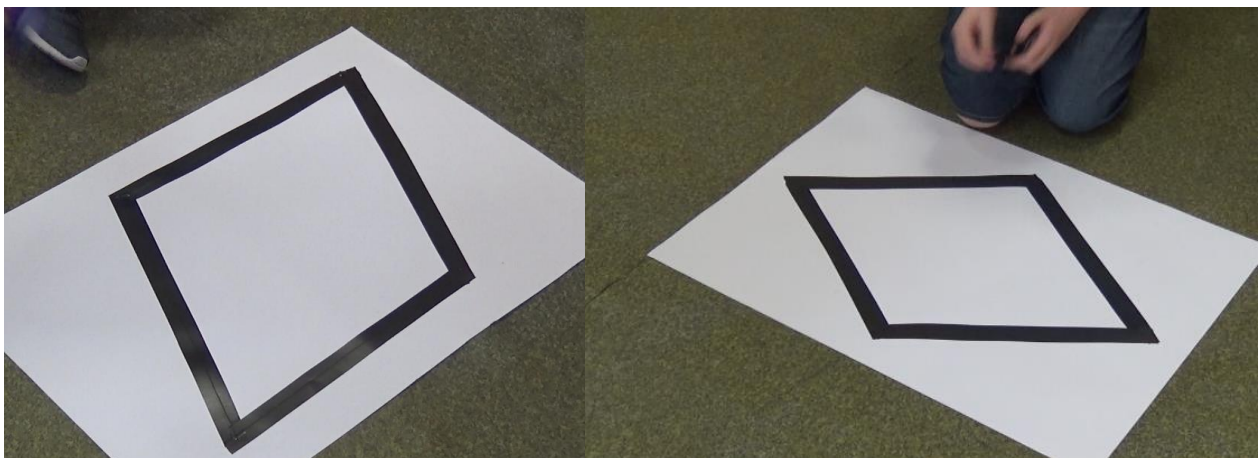
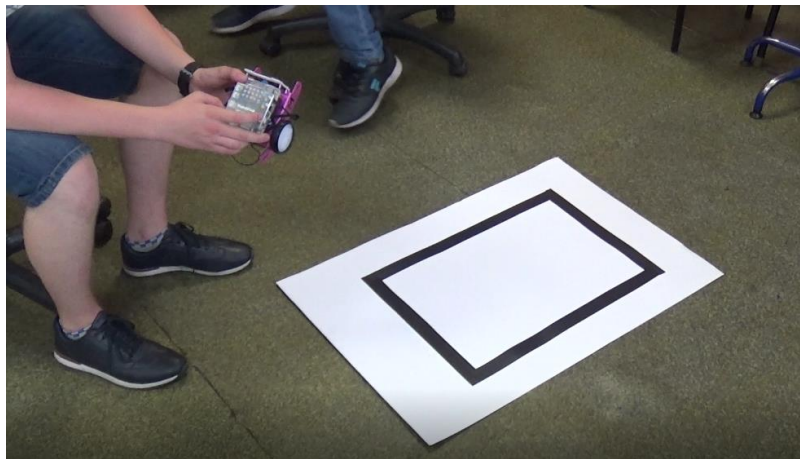
### Versión 1:

Se les reparten a los estudiantes unas hojas con una figura geométrica rectangular.

Los alumnos disponen de datos de velocidades obtenidos para distintas potencias de los motores. (Estos datos los han obtenido en la actividad 7 de la guía, "7-Cálculo de velocidades").

Potencia motores	Velocidad media [cm/s]
50	5,1
100	12,9
150	20,1
200	24,9
255	28,7

El robot nos da el tiempo que tarda en recorrer un lado y luego el otro lado. A continuación el programa nos pide estos datos para poder calcular el área de la figura.

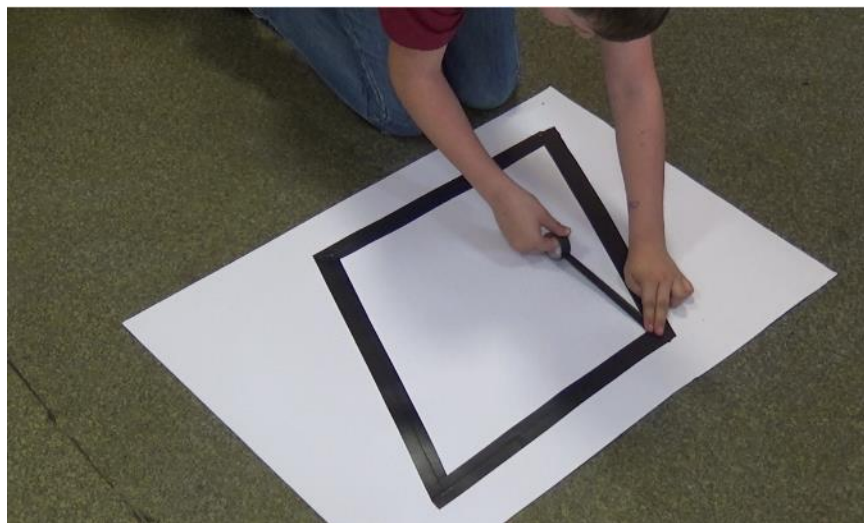
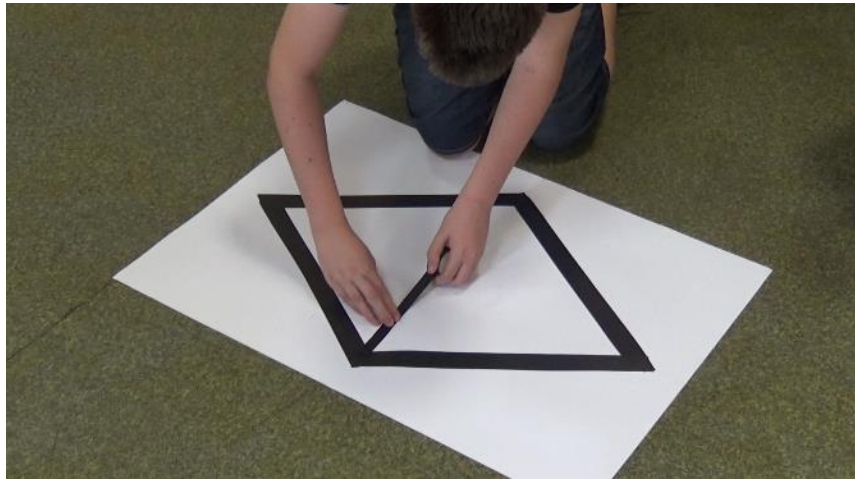


Para cada figura con la que trabajemos, somos nosotros los que fijamos la velocidad y la introducimos en el programa. (El programa nos la pide).

A continuación introducimos en el programa, los tiempos que tarda el robot en recorrer cada lado del rectángulo. El programa utiliza estos tiempos para calcular la longitud de los lados del rectángulo.

Por último, el programa, teniendo todos los datos que necesita, aplica la fórmula y calcula el área de la figura.

La altura o las diagonales se utilizan para rombos y paralelogramos. Los estudiantes pueden colocar / pegar las piezas faltantes de la figura.



El siguiente programa nos muestra ambas partes del programa:

- La parte en que se calculan los tiempos que el robot tarda en recorrer los lados. El tiempo se muestra en la matriz de leds. (Este programa lo podemos subir a la placa del robot. En este caso, para el sensor Sigue Líneas, el color blanco significa que el robot se para).
- La parte en la que se calcula el área de la figura, que se ejecuta, al apretar la tecla "p" del teclado del ordenador.



```

mBotProgram
run forward at speed 0
wait until on board button pressed
wait until on board button released
reset timer
set see to line follower Port2
repeat until see = 3
  if see = 1 then
    set motor M1 speed 0
    set motor M2 speed 255
  else
    if see = 2 then
      set motor M1 speed 255
      set motor M2 speed 0
    else
      if see = 0 then
        set motor M1 speed 150
        set motor M2 speed 150
      else
        set motor M1 speed 0
        set motor M2 speed 0
        set time to timer
        show face Port1 number time
  end if
end if
end repeat

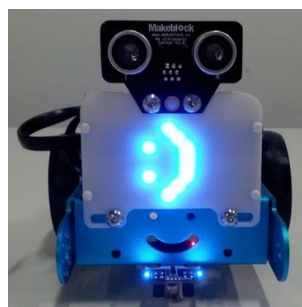
when p key pressed
say The area of rectangle
ask Give the speed [cm/s] and wait
set v to answer
ask Give the time of passage on first side and wait
set a to answer
ask Give the time of passage on second side and wait
set b to answer
set ad to a * v
set bd to b * v
say ad * bd

```

Para otras figuras, solo se modifica la segunda parte del programa. Modificaríamos lo que nos pregunta el programa y las distintas fórmulas.

**Versión 2:**

Comenzamos añadiendo a la actividad, el sensor de Ultrasonidos:



Desarrollamos el código para el **CUADRADO**, que es el más fácil:

```
when clicked
  show face Port4 x: 0 y: 0 characters:
  set led on board all red 0 green 0 blue 150
  set area_edge to 0
  set area_TOTAL to 0
  Question_Edge
```

```
define Question_Edge
  set led on board all red 0 green 0 blue 150
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: How many cm measures the edge?
    change xPos_text by -1
    wait 0.0005 secs
    if xPos_text < -180 then
      Save Answer EDGE
```

```
define Save Answer EDGE
  set led on board all red 255 green 0 blue 0
  forever
    if ultrasonicsensor Port3 distance > 390 then
      show face Port4 number 0
    else
      show face Port4 number 1.25 * ultrasonicsensor Port3 distance
      if on board button pressed then
        set area_edge to 1.25 * ultrasonicsensor Port3 distance
        Calculate_Area
```



```

define Calculate_Area
  set led on board all red 0 green 150 blue 150
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: The area of square is
    change xPos_text by -1
    wait 0.0005secs
    if <xPos_text < -130 then
      Answer
  
```

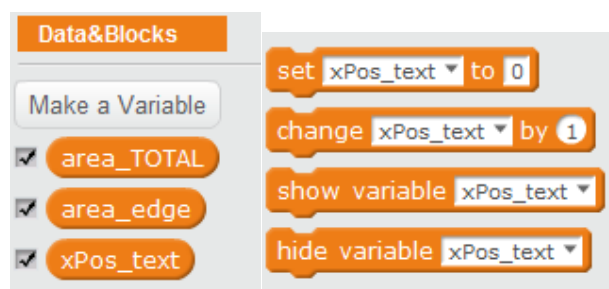
```

define Answer
  set led on board all red 0 green 0 blue 150
  set area_TOTAL to area_edge * area_edge
  forever
    show face Port4 number: area_TOTAL
  
```

Para poder hacer este programa, hemos creado 3 variables y 4 bloques:

Variables:

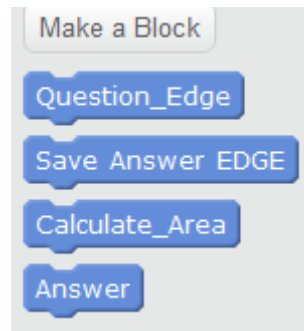
- ❖ Área total (*area\_TOTAL*) => se utiliza para mostrar el valor de toda el área.
- ❖ Borde del área (*area\_edge*) => se usa para guardar el valor de un borde de la figura.
- ❖ *xPos\_text* => se utiliz para que podamos mover el texto que aparece en la Matriz Led.



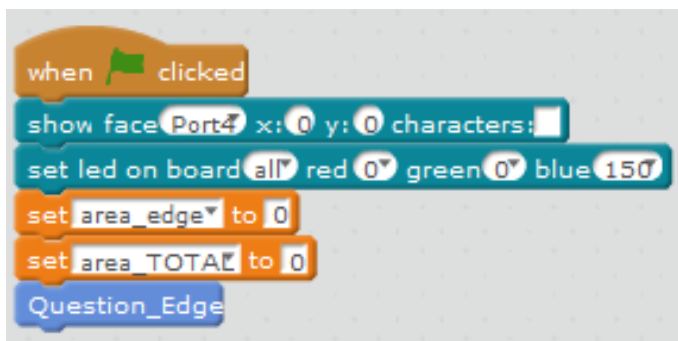
Bloques:

- ❖ “Preguntar borde de la figura”, (*Question\_Edge*) => mediante este bloque, el mBot nos pide que calculemos la distancia de un borde.
- ❖ “Guardar respuesta borde”, (*Save answer EDGE*) => este bloque se utiliza para guardar el valor que el mBot calcula de la variable “borde del área”.

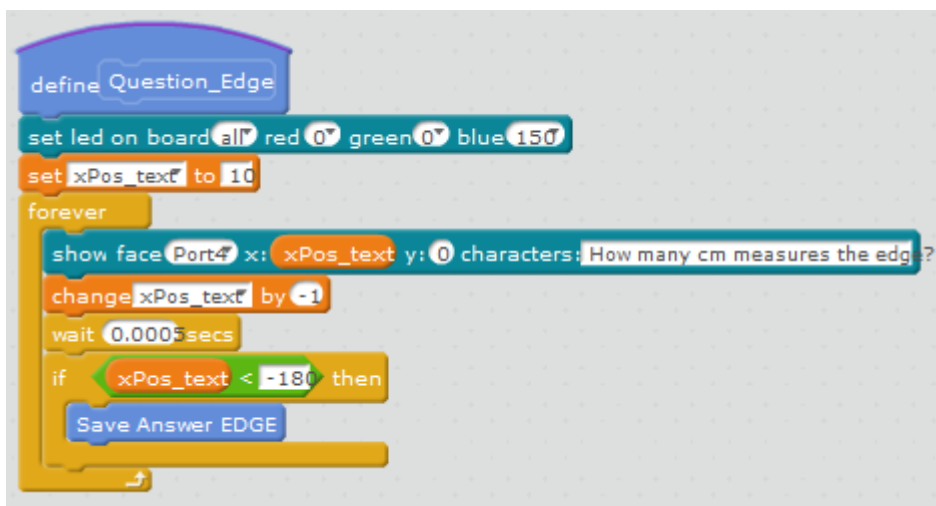
- ❖ “Calcular área”, (*Calculate\_Area*) => mediante este bloque el mBot nos anuncia que va a calcular el área.
- ❖ “Respuesta”, (*Answer*) => con este bloque el mBot calcula el área y muestra el resultado en la matriz de leds.



Veamos cada uno de los bloques:



Al darle a la bandera, se inicia el programa, se ponen las variables a 0 y pasamos al bloque “Preguntar borde de la figura”.



Inicializamos el "xPos\_text" a 10 porque la primera letra de la pregunta aparece a la derecha. Cada 0.0005 las letras se moverán a través de la matriz de Leds. Cuando se haya visualizado toda la pregunta, pasaremos al siguiente bloque.

```

define Save Answer EDGE
set led on board all red 255 green 0 blue 0
forever
if ultrasonicsensor Port3 distance > 390 then
show face Port4 number 0
else
show face Port4 number 1.25 * ultrasonicsensor Port3 distance
if on board button pressed then
set area_edge to 1.25 * ultrasonicsensor Port3 distance
Calculate_Area

```

Para evitar posibles confusiones, el mBot cuando no detecta nada se muestra en el conjunto de LED 400, por lo tanto, para evitarlo, indicamos que si la distancia es mayor que 390, muestra 0.

Cuando la distancia es menor que 390, podemos guardar en la variable "borde del área" la distancia que se calcula en ese momento. Para ello, hay que presionar el botón en la placa del robot.

**Nota:** se puede ver en el programa que la distancia que calcula el sensor de Ultrasonidos se multiplica por 1.25. Esto se debe a que el sensor de ultrasonidos tiene una desviación. Al aplicar este factor de corrección, obtenemos el cálculo correcto. En la sección "Más información", explicamos esto más detalladamente.

```

define Calculate_Area
set led on board all red 0 green 150 blue 150
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: The area of square is
change xPos_text by -1
wait 0.000 secs
if xPos_text < -13 then
Answer

```

El mBot muestra en la matriz de leds "eEl área del cuadrado es...".

```

define Answer
set led on board all red 0 green 0 blue 150
set area_TOTAL to area_edge * area_edge
forever
show face Port4 number: area_TOTAL

```

Dado que el área del cuadrado es  $\text{lado}^2$ , el programa realiza este cálculo y el mBot muestra el resultado obtenido.

Para calcular el perímetro, utilizaremos el mismo programa, salvo que en el bloque "Respuesta", el borde se multiplicará por 4.

```

when clicked
  show face Port4 x: 0 y: 0 characters:
  set led on board all red 0 green 0 blue 150
  set perimeter_edge to 0
  set perimeter_TOTAL to 0
  Question_Edge
  
```

```

define Question_Edge
  set led on board all red 0 green 0 blue 150
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: How many cm measures the edge?
    change xPos_text by -1
    wait 0.000 secs
    if xPos_text < -18 then
      Save Answer EDGE
  
```

```

define Save Answer EDGE
  set led on board all red 255 green 0 blue 0
  forever
    if ultrasonicsensor Port3 distance > 390 then
      show face Port4 number 0
    else
      show face Port4 number 1.25 * ultrasonicsensor Port3 distance
      if on board button pressed then
        set perimeter_edge to 1.25 * ultrasonicsensor Port3 distance
        Calculate_Perimeter
  
```

```

define Calculate_Perimeter
  set led on board all red 0 green 150 blue 150
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: The perimeter of square is
    change xPos_text by -1
    wait 0.000 secs
    if xPos_text < -15 then
      Answer
  
```

```

define Answer
  set led on board all red 0 green 0 blue 150
  set perimeter_TOTAL to perimeter_edge * 4
  forever
    show face Port4 number: perimeter_TOTAL
  
```

Si lo que queremos es calcular el área de un **TRIÁNGULO**, el mBot primero nos preguntará por el valor de la base y de la altura de dicho triángulo. Luego calculará el área.

```

when clicked
  show face Port4 x: 0 y: 0 characters: 
  set led on board all red 0 green 0 blue 150
  set area_base to 0
  set area_height to 0
  set area_TOTAL to 0
  Question_Area_Base

define Question_Area_Base
  set led on board all red 0 green 0 blue 150
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: How many cm does the base measure
    change xPos_text by -1
    wait 0.000 secs
    if <xPos_text < -21 then
      Save Answer BASE
  
```

```

define Save Answer BASE
  set led on board all red 255 green 0 blue 0
  forever
    if ultrasonicsensor Port3 distance > 390 then
      show face Port4 number 0
    else
      show face Port4 number 1.25 * ultrasonicsensor Port3 distance
      if on board button pressed then
        set area_base to 1.25 * ultrasonicsensor Port3 distance
        Question_Area_Height
  
```

```

define Question_Area_Height
  set led on board all red 0 green 150 blue 0
  set xPos_text to 10
  forever
    show face Port4 x: xPos_text y: 0 characters: How many cm is the height
    change xPos_text by -1
    wait 0.000 secs
    if <xPos_text < -16 then
      Save Answer HEIGHT
  
```

```

define Save Answer HEIGHT
set led on board all red 255 green 0 blue 0
forever
if ultrasonicsensor Port3 distance > 390 then
show face Port4 number 0
else
show face Port4 number 1.25 * ultrasonicsensor Port3 distance
if on board button pressed then
set area_height to 1.25 * ultrasonicsensor Port3 distance
Calculate_Area

```

```

define Calculate_Area
set led on board all red 0 green 150 blue 150
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: The area of triangle is:
change xPos_text by -1
wait 0.000 secs
if xPos_text < -15 then
Answer

```

```

define Answer
set led on board all red 0 green 0 blue 150
set area_TOTAL to area_base * area_height / 2
forever
show face Port4 number: area_TOTAL

```

Y para calcular el perímetro del triángulo, el mBot preguntará cuántos cm mide cada borde y luego calculará el perímetro total:

```

when clicked
show face Port4 x: 0 y: 0 characters:
set led on board all red 0 green 0 blue 150
set perimiter_edge1 to 0
set perimiter_edge2 to 0
set perimiter_edge3 to 0
set perimeter_TOTAL to 0
Question_Perimeter_Edge

```

```

define Question_Perimeter_Edge1
set led on board all red 0 green 0 blue 150
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: How many cm measures the first edge?
change xPos_text by -1
wait 0.0005 secs
if xPos_text < -22 then
Save Answer_EDGE1

```

```

define Save Answer_EDGE1
set led on board all red 255 green 0 blue 0
forever
if ultrasonicsensor Port3 distance > 390 then
show face Port4 number 0
else
show face Port4 number 1.25 * ultrasonicsensor Port3 distance
if on board button pressed then
set perimeter_edge1 to 1.25 * ultrasonicsensor Port3 distance
Question_Perimeter_Edge1

```

```

define Question_Perimeter_Edge2
set led on board all red 0 green 150 blue 0
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: How many cm is the second edge?
change xPos_text by -1
wait 0.0005 secs
if xPos_text < -16 then
Save Answer_EDGE2

```

```

define Save Answer EDGE2
set led on board all red 255 green 0 blue 0
forever
if ultrasonicsensor Port3 distance > 390 then
show face Port4 number 0
else
show face Port4 number 1.25 * ultrasonicsensor Port3 distance
if on board button pressed then
set perimeter_edge2 to 1.25 * ultrasonicsensor Port3 distance
Question_Perimeter_Edge3

```

```

define Question_Perimeter_Edge3
set led on board all red 0 green 150 blue 0
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: How many cm is the third edge?
change xPos_text by -1
wait 0.005secs
if xPos_text < -16 then
Save Answer EDGE 3

```

```

define Save Answer EDGE 3
set led on board all red 255 green 0 blue 0
forever
if ultrasonicsensor Port3 distance > 390 then
show face Port4 number 0
else
show face Port4 number 1.25 * ultrasonicsensor Port3 distance
if on board button pressed then
set perimeter_edge3 to 1.25 * ultrasonicsensor Port3 distance
Calculate_Perimeter

```



```

define Calculate_Perimeter
set led on board all red 0 green 150 blue 150
set xPos_text to 10
forever
show face Port4 x: xPos_text y: 0 characters: The perimeter of triangle
change xPos_text by -1
wait 0.000 secs
if <xPos_text < -15> then
Answer

```

```

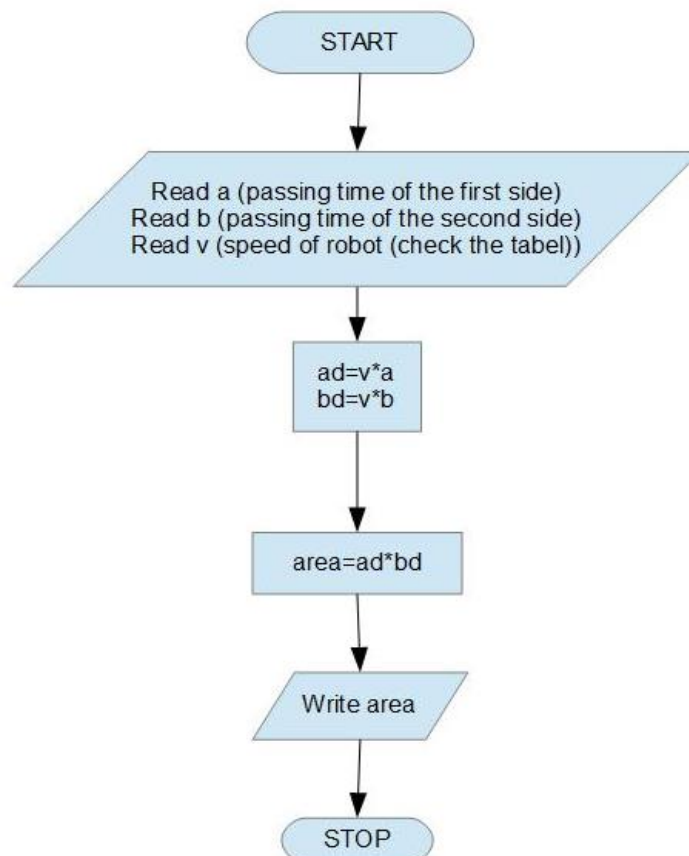
define Answer
set led on board all red 0 green 0 blue 150
set perimeter_TOTAL to perimeter_edge1 + perimeter_edge2 + perimeter_edge3
forever
show face Port4 number: perimeter_TOTAL

```

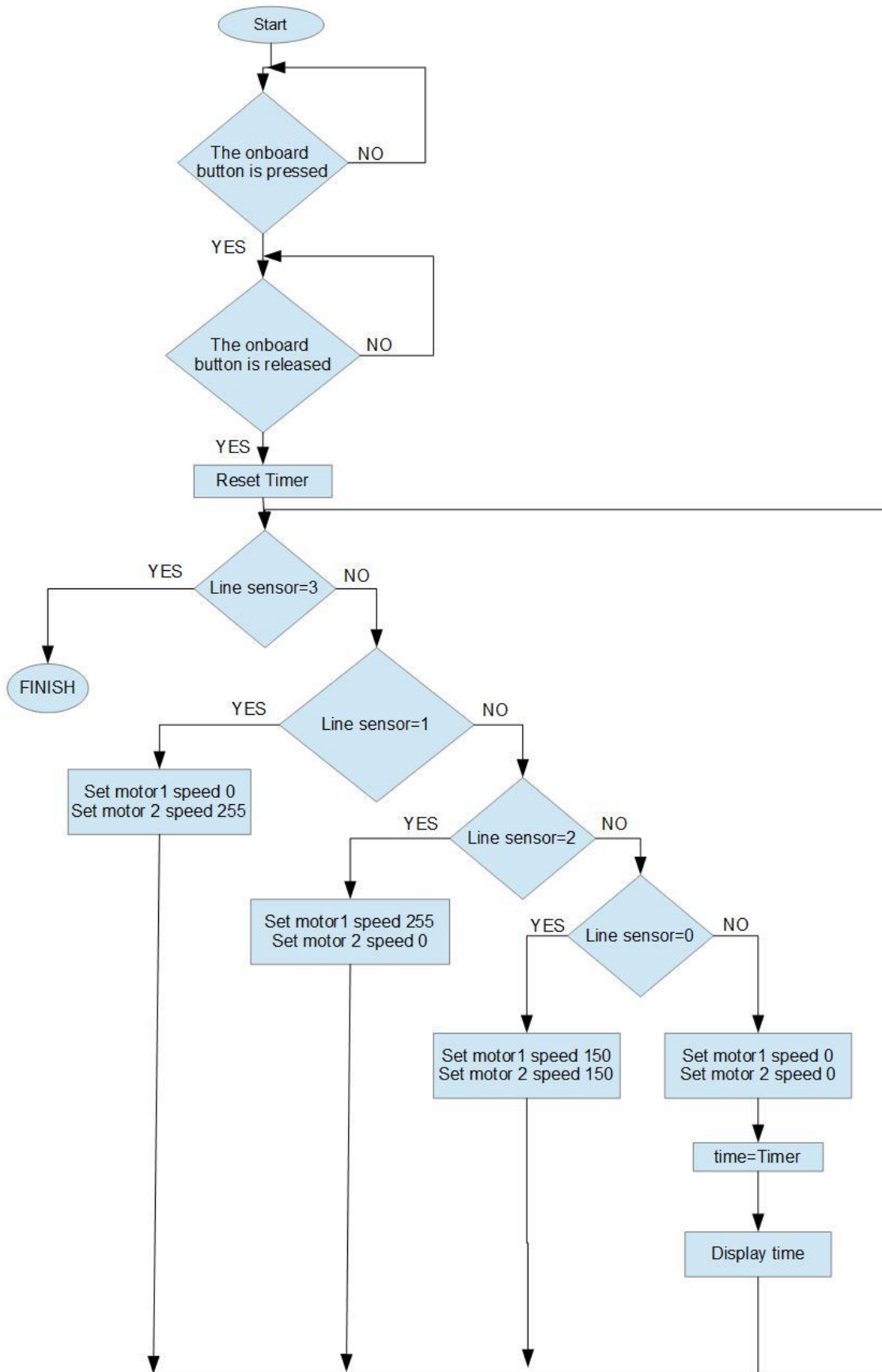
### DIAGRAMA DE FLUJO

#### Versión 1:

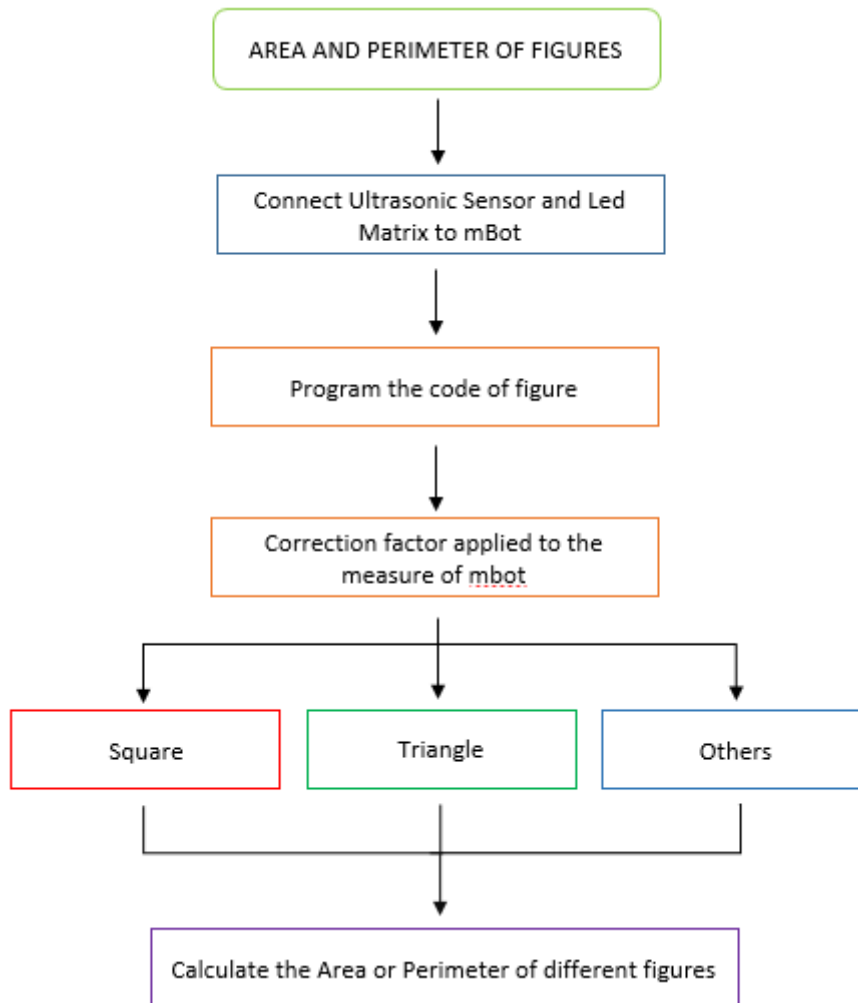
Programa para el cálculo del área de un rectángulo:



# Programa para medir el tiempo:



## Versión 2:



## EVALUACIÓN DE LOS ESTUDIANTES

### Versión 1:

Podemos evaluarles, haciendo que utilicen la fórmula para calcular el camino, cuando conocen la hora y la velocidad. También podemos pedirles que escriban el programa para calcular el área de otras figuras como trapecio o deltoides.

### Versión 2:

Para evaluar la competencia del estudiante, primero, el profesor debe enseñar diferentes figuras geométricas, y cómo se calcula el área y el perímetro de cada una. Luego deben hacer esas figuras en 3D o imprimirlas en un papel, y ya, con la ayuda del mBot, poner en práctica la teoría dada por el profesor.

Después de todo este proceso, sería también aconsejable, realizar un examen práctico, sin la ayuda del mBot, para verificar que el estudiante haya entendido la geometría.



## BIBLIOGRAFÍA

<http://www.makeblock.com/>

<https://makeblock.es/>

## ESCALABILIDAD

La escalabilidad de esta actividad, consistiría en programar diferentes figuras geométricas cada vez más complejas.

## INFORMACIÓN ADICIONAL

Con la ayuda de un medidor, nos dimos cuenta de que el sensor de ultrasonido medía menos que la medida real. Decidimos entonces hacer una tabla con la medida real y con la medida calculada por el mBot; de esta manera podríamos utilizar ambos pares de datos para calcular un factor de corrección con el fin de hacer la medida del mBot lo más precisa posible.

Real Measure (cm)	MBOT measure (ultrasonic sensor)	Difference
10	8,5	1,5
20	15,1	4,9
30	23,4	6,6
40	30,8	9,2
50	39,6	10,4
60	46,9	13,1
70	53,4	16,6
80	62,6	17,4
90	70,3	19,7

