

*Este documento es una traducción de la actividad de “My Robotic Friends” desarrollada por Thinkersmith y publicada en abierto para la Computer Science Education Week 2013. Más información en <http://csedweek.org/unplugged/thinkersmith> No se trata de una traducción literal y el autor ha añadido algunas anotaciones para docentes que quieran llevar a cabo esta actividad en clase, alimentadas de su propia experiencia poniéndola en práctica.*

# Mis Amigos Robot

*Israel Gutiérrez (@gootyfer)*

La actividad de mis amigos robot es una actividad que permite introducir a no iniciados, normalmente niños aunque también funciona con adultos, en el mundo de la programación, dejando claro algunos conceptos como programar, algoritmo o código fuente.

## Objetivo Principal

Destacar técnicas de programación e ilustrar la necesidad de funciones.

## Resumen

Usando un vocabulario predefinido los estudiantes escribirán un programa para construir una estructura con vasos de plástico, que otro de los estudiantes ejecutará a modo de robot. Esta actividad servirá para introducir conceptos de programación, por ejemplo depuración de errores (debugging) ya que la actividad está diseñada para ser ambigua en algunos aspectos y que los alumnos se equivoquen. Si el tiempo lo permite, al final existe una parte final para introducir el concepto de función.

## Objetivos

Los alumnos...

- aprenderán a convertir actividades del mundo real en instrucciones
- practicarán codificación de instrucciones con símbolos
- entenderán la necesidad de precisión a la hora de programar
- practicarán depuración de errores de código no funcional
- entenderán la utilidad de funciones y parámetros (a partir de 12 años)

## Metodología

Los alumnos trabajarán en grupos, con unos objetivos y distribución que depende de su edad:

- de 5 - 8 años, ejecutarán la lección todos juntos como una clase, indicando entre todos al profesor las instrucciones del programa
- de 9 - 11 años, en grupos de 3-5 alumnos en los que cada uno pasará por los roles de programador y robot
- a partir de 12 años trabajarán en grupos de 3 alumnos, dos siendo programadores y uno el robot (por turnos); también podrán realizar la parte extra sobre funciones

## Materiales necesarios

Cada grupo de alumnos necesitará

- Una impresión de las páginas 11-14 del [manual de Thinkersmith](#) (hoja de símbolos y muestra de construcciones con vasos) para cada grupo participante (normalmente de 3 alumnos, más información después)
- 17 vasos de plástico
- Papel y boli

## Espacio y personal

Lo ideal es realizar la actividad en un espacio que permita el profesor y ayudantes pasar fácilmente a ayudar a los grupos con dudas durante el desarrollo. Además, se tendrá una zona apartada para que los robots practiquen ejecuciones mientras esperan a que los programadores terminen de escribir el código. Idealmente se requiere un profesor o ayudante por cada 4-5 grupos de alumnos.

## Descripción de la Actividad

### Vocabulario

- Algoritmo: conjunto ordenado de instrucciones que permite completar una tarea
- Codificar / programar: transformar acciones es un lenguaje de símbolo entendible por un ordenador
- Depuración de errores (debugging): encontrar y solucionar problemas en el código
- Función: trozo de código que se repite, y puede ser llamado una y otra vez
- Parámetros: información extra que se pasa a las funciones para personalizarlas

### Introducción

Empezar preguntando si algún chico ha escuchado hablar sobre robótica. Alguno ha visto un robot o lo ha tocado? El robot realmente te escucha y entiende lo que dices? Respuesta: no de la misma forma que una persona.

Los robots operan en base a una serie de **instrucciones**, un conjunto de cosas que están preprogramados para realizar. Para completar una tarea el robot necesita una serie ordenada de instrucciones (normalmente llamado **algoritmo**) que puede ejecutar. Hoy vamos a aprender cómo se hace todo esto.

### Ejecución

Saca la hoja con los símbolos (o escríbelos en la pizarra, ponlos en el proyector). Indica claramente qué estos serán los **únicos 6 símbolos** que podrán usar para escribir el programa. Este programa será ejecutado por *su robot* para construir una estructura de vasos usando sólo estas flechas:

- ↑ Cogeer vaso (no subir vaso)
- ↓ Soltar vaso (no bajar vaso)
- Mover  $\frac{1}{2}$  vaso a la derecha
- ← Mover  $\frac{1}{2}$  vaso a la izquierda
- ↻ Girar el vaso 90° a la derecha
- ↺ Girar el vaso 90° a la izquierda

### Pasos

1. Elegir un robot por equipo
2. Enviar el robot a la biblioteca mientras los programadores programan
3. Elegir una estructura de vasos para cada equipo
4. Los programadores del equipo crearán un algoritmo para construir la estructura
5. Los programadores traducirán su algoritmo en flechas (símbolos) escribiendo en el papel
6. Cuando los programadores hayan terminado, pueden llamar a su robot
7. Cuando el robot llega ejecutará el programa sin hablar con el equipo de programadores, y sin conocer la estructura a construir
8. Los programadores observarán movimientos incorrectos, y si hay un error tendrán que depurarlo y pedir al robot que vuelva a ejecutar el programa

### Reglas

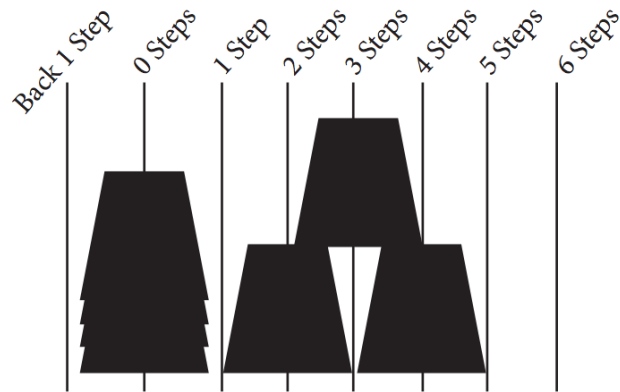
- Los programadores sólo pueden usar las **6 flechas** para construir el programa
- Los **vasos deben estar con el robot**, no con los programadores mientras programan
- Cuando los robots vuelven al grupo a ejecutar un programa, **no pueden hablar** con el resto del grupo

### Ejemplo

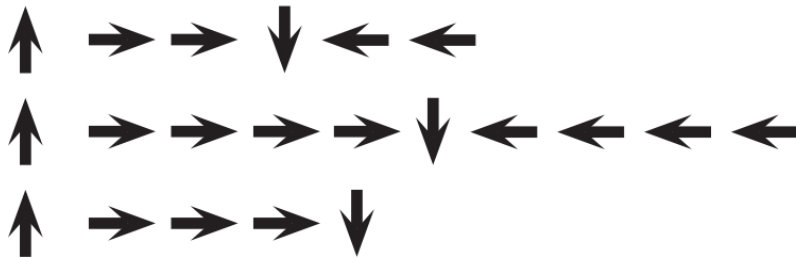
Es bastante útil empezar con un ejemplo con toda la clase, normalmente uno sencillo con 3 vasos. Muestra el modelo a la clase y guíales en el ejercicio.



Coloca una pila de vasos sobre la mesa, donde todo el mundo pueda verla, y pide a la clase que te indique qué símbolos usar. El primer es “coger vaso”. Cuando coges un vaso ten en cuenta que al moverse **pasará automáticamente por encima de cualquier vaso** de la estructura. A continuación pide el resto de instrucciones, recordando que las flechas horizontales sólo mueven medio vaso de distancia.



Un ejemplo del programa para conseguirlo se indica a continuación. Nota que hay que usar flechas de retorno para volver a la pila a coger un nuevo vaso. Y que se ha usado una línea por vaso para facilitar la lectura del código.



### Comienza el espectáculo

1. Pon a los alumnos por **grupos**

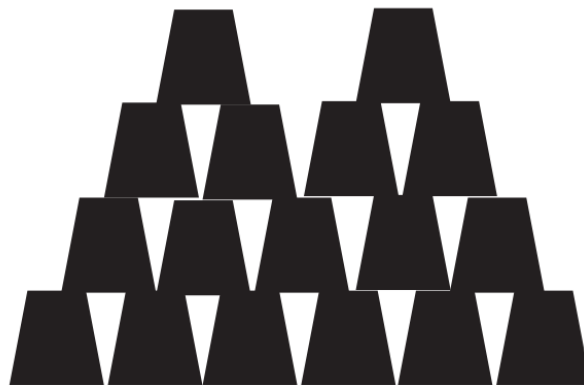
2. Elige un **robot**, que irá rotando en siguientes ejercicios y que se marchará a practicar ejecuciones con los vasos a la biblioteca
3. Dar a los **programadores** los materiales necesarios, e indicarles una estructura de vasos a implementar. Que empiecen a pensar en el algoritmo para conseguir la estructura, y cuando lo consigan que escriban el programa en el papel usando únicamente los símbolos indicados. Deberán revisar el programa antes de pedir al robot que ejecute.
4. El robot **ejecuta el programa** sin hablar con sus compañeros, centrándose en el programa que le han pasado en la hoja. Si los programadores encuentran un fallo en el programa podrán abortar su ejecución, enviar al robot de vuelta a la biblioteca y solucionar el error antes de pedirle que lo ejecute de nuevo (introducir concepto de depuración de errores).
5. Cuando el equipo solucione un reto, **repetir** con un nuevo robot y una estructura de vasos más compleja.

## Suplemento de Funciones

Si tenemos tiempo extra en la sesión tendremos la oportunidad de presentar la utilidad de las funciones. Esta parte está especialmente indicada para los niños más mayores (a partir de 12 años). Supondrá aproximadamente los últimos 15-20 minutos de la sesión.

### Ejecución

Junta a todos los estudiantes y diles que vas a darles un estructura de vasos especial y que quieres que la programen en un tiempo record. Luego enséñales esto:



Lo normal es que escuches quejas, resoplidos e incluso expresiones de rechazo. Pregunta que cuál es el problema. ¿Qué hace esta estructura tan complicada?

La estructura es compleja porque cada vaso adicional a lo ancho de la estructura añade cuatro flechas más en el programa (2 a la derecha y dos a la izquierda).

Pide a los estudiantes que sean creativos, y algunos de ellos comenzarán a escribir versiones acortadas de las instrucciones, por ejemplo

$$\uparrow \rightarrow(12) \downarrow \leftarrow(12)$$

Durante el juego normal este método debe ser advertido y desaconsejado: recuérdales que deben ceñirse a los 6 símbolos de las flechas. En esta parte, en cambio, reconoce la brillantez de los que lo hagan, y reconoce que ¡ellos descubrieron la necesidad de funciones de forma independiente!

Explica a la clase que las flechas con números son una forma inteligente de indicar que queremos repetir la flecha un número específico de veces. Permitiendo esto, esencialmente estamos creando un **nuevo símbolo que podemos usar para evitar repetir código innecesariamente**. Esa es exactamente la idea detrás de las funciones.

Anima a la clase a encontrar trozos grande de código repetido. Como instructor puedes indicar cualquier agrupación que tenga sentido, pero debería ser algo parecido a esto:

$$\uparrow \rightarrow(X) \downarrow \leftarrow(X)$$

Es un buen punto intermedio pero podemos simplificar incluso más. Si colocamos los símbolos en el sentido de las agujas del reloj (empezando por arriba), podemos convertirlos en un único símbolo, que podría ser algo como esto, donde la X es el número de pasos a mover hacia delante y luego hacia atrás. X es por tanto un parámetro de la función.



### Parámetros

En este último símbolo hemos incluido una forma de **pasarle cierta información** a cerca de cuántas veces queremos aplicar las flechas de movimiento a derecha e izquierda. En el mundo de la programación esa información extra se llama parámetro y sirve para personalizar una función.

Ahora que tenemos una nueva simbología, pídeles afrontar algún reto más intenso, de las estructuras de 17 vasos o alguna más compleja que ellos mismo inventen.