

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Control de autenticación y
 acceso (Spring Security)
 UDDI

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Web Services
 Rest Services
 Social SSO
 SSO (Cas)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



Entra en Adictos a través de

E-mail

Contraseña

Entrar [Registrarme](#)
[Olvidé mi contraseña](#)

[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestros libros](#) [Más](#)

» [Estás en: Inicio](#) [Tutoriales](#) [Taller Lightbot programación para niños.](#)



Isabel Rodríguez Olivar

Directora desarrollo de Negocio

Titulada en Ingeniería Técnica en Informática por la Universidad Carlos III de Madrid.

Con más de 15 años de experiencia en las TIC, provengo de la parte técnica donde he desarrollado el trabajo en diferentes perfiles, desde el 2006 me he dedicado a la parte comercial y de gestión

Twitter: [Seguir a @iro2009](#) 161 seguidores

Actualmente me podéis encontrar en Autentia: Ofrecemos servicios de soporte a desarrollo, factoría y formación
Somos expertos en Java/J2EE

[Ver todos los tutoriales del autor](#)



Fecha de publicación del tutorial: 2014-09-04

Tutorial visitado 1 veces [Descargar en PDF](#)

Taller Lightbot programación para niños.

El objetivo principal es enseñar a los niños y personas que no han programado nunca, una visión general de lo que es Lightbot y lo que tiene que ver con la programación Aprendizaje y Ciencias de la Computación.

Sobre el autor:

Danny Yaroslavski ha trabajado en compañías como Electronic Arts, Armor Games Inc, y Side FX Software. Él es el desarrollador principal de Lightbot y tiene una Licenciatura en Ciencias de la Computación de la Universidad de Waterloo (Canadá).

¿Qué es la programación? (Concepto general)

La programación es la forma en que una persona le dice a un ordenador qué hacer. Un ordenador no entiende un "Inglés o español". A un ordenador no se le puede decir "ve a pasear al perro" o "crea una pintura de la Mona Lisa". Los ordenadores comprenden sólo un pequeño conjunto de instrucciones. La programación es la manera de resolver un problemas a partir de una instrucciones básicas que entiende.

La visión que en general que se tiene de los programadores y la programación es que pueden escribir un lenguaje que entiende el ordenador (C ++, Java, Python) y el resultado de ejecutarlo. Por ejemplo:

```
int a = 5 ;
int b = a + 5;
print ( b );
Output: 10
```

Sin embargo, lo más importante de la programación, no es conocer el lenguaje que el ordenador entiende, o cómo escribir expresiones como se muestran arriba, sino el proceso con el que llegamos a una solución y pensar algorítmicamente sobre cómo resolver un problema.

¿Qué es Lightbot?

Lightbot es un juego de puzzle de programación. Esto significa que en su núcleo, es un juego de rompecabezas, pero su mecánica de juego hace que nos familiaricemos con los conceptos básicos de la programación.

El objetivo de Lightbot es:

En cada nivel, guiar a un robot para iluminar todos los azulejos azules. Para ello, debe 'programar' el robot usando un conjunto de instrucciones. Por ejemplo:

flecha: Un icono de la flecha indica al robot para avanzar un espacio.

bombilla: Un símbolo de la bombilla indica al robot que ilumine el azulejo en el que está.

Relacionar Lightbot a la Programación

Si entendemos la programación como pensar algorítmicamente para llegar a una solución, podemos ver cómo Lightbot te ayuda a pensar de esa manera.

A continuación, vamos a ver los conceptos que se enseñan en Lightbot. Estos conceptos se dividirán en dos grupos, el segundo de los cuales tiene un vínculo más "directo" a la programación.

Catálogo de servicios Autentia



Síguenos a través de:



Últimas Noticias

» Portales, gestores de contenidos documentales y desarrollos a medida

» Comentando el libro Start-up Nation, La historia del milagro económico de Israel, de Dan Senor & Salu Singer

» [Screencasts de programación narrados en Español](#)

» [Sorteo de entradas para APIdays Mediterranea](#)

» [Concurso del Día de la Madre:](#)

[Histórico de noticias](#)

Últimos Tutoriales

» [Mockear la capa back con Dyon](#)

» [Automatizando los Smoke Test con TestLink y Jenkins](#)

» [Cómo hacer testing automático de un applet Java](#)

» [Primeros pasos con Clojure: Leiningen y Midje](#)

» [Smoke Test implementados con TestNG y Selenium](#)

Prácticas de programación

Planificación

Al principio, los jugadores deben evaluar las instrucciones que tienen disponibles para ellos. Los jugadores deben ponerse en la posición del robot y encontrar la manera de guiar al robot para resolver el nivel.

Esto es lo comparable a los programadores cuando tienen que desarrollar un programa, deben entender y visualizar un problema que tienen y evaluar las instrucciones que tienen disponibles en su idioma de elección, y crear un plan de acción.

Programación

Los jugadores deben secuenciar las instrucciones de acuerdo a su plan (mediante iconos de instrucciones).

Esta es la misma forma como los programadores deben escribir su programa (usando el código).

Pruebas

Los jugadores ejecutan el programa que han creado y prueban si el resultado es el deseado. Esto es la misma cuando se ejecuta de una pieza de código y se ve si el resultado se mantiene con lo que se espera.

Depuración

Cuando un nivel que no se resuelve correctamente, los jugadores deben buscar lo que puede haber causado el problema. Esto se puede hacer recorriendo los comandos que hemos puesto e incluir, modificar o eliminar comandos para entender donde está el problema. Esto es lo mismo que lo que los programadores tienen que hacer cuando hay un error en su código. Por lo general, tienen que volver a ejecutar el programa paso a paso, prestando especial atención a lo que pudo haber salido mal, recorriendo las instrucciones para poder encontrar donde se produce el error.

Procedimientos

Jugadores de Lightbot deben utilizar procedimientos cuando no tienen suficiente espacio en el bloque principal para resolver un problema. Los procedimientos son útiles para la extracción de patrones y re-uso de un conjunto de comandos varias veces. Los procedimientos también se ejecutan de una manera que, cuando se ejecuta un icono de P1, se ejecutan todos los comandos en PROC1, y al final de PROC1, la ejecución vuelve al comando siguiente del icono de P1.

Esto es muy similar a los procedimientos o funciones en un lenguaje de programación. Los procedimientos son útiles para la reutilización de código y extraer fuera código, intentar no duplicar código, ya que en posteriores modificaciones solo tienes que hacerlas en un solo sitio.

Bucles

Jugadores en Lightbot pueden utilizar bucles para resolver algunos niveles. Los bucles son útiles para tareas repetitivas, y son una extensión de la extracción de un patrón que se produce una y otra vez.

En Lightbot el tipo de bucle que se utiliza es el recursivo. Esto es muy parecido a un bucle recursivo en un lenguaje de programación, útiles para ejecutar código que ocurre una y otra vez, se llama a un procedimiento desde dentro el procedimiento en sí (es decir. Poner P1 en PROC1). Al final del bucle, la ejecución del programa salta al principio del bucle, como lo hace en el juego.

Contenidos del taller Lightbot

Objetivo principal: Comprender cómo crear y dar al ordenador una serie de instrucciones a seguir.

Creación y ejecución de un Programa

Los estudiantes jugarán el conjunto inicial de los niveles "básicos" en Lightbot. Aprenderán cómo decirle a un ordenador qué hacer con una serie de comandos básicos y hacerse con el proceso general.

¿Qué se necesita?

Juego Lightbot (en un ordenador o dispositivo iOS o Android)

Pizarra / proyector

Objetivos del taller

- 1) Planificar una serie de instrucciones
- 2) Escribir un programa de Lightbot y ejecutar a través de las instrucciones
- 3) Probar si el resultado final fue correcta (todos los azulejos azules que se encendían)
- 4) Depurar Si el resultado no es el correcto, recorrer las instrucciones para ver que pasó, encontrar el error, y modificar el programa

En el taller de lightboot se jugará a los 3 primeros niveles siguiendo las instrucciones del juego.

Contenido

Se trata de superar los "Fundamentos" nivel establecido. Para controlar el robot se necesita darle un conjunto de instrucciones, llamado a un "programa", para funcionar. Una vez que se pulsa el botón de reproducción verde(), el robot llevará a cabo todas las instrucciones en el programa. Si se comete un error, se tendrá que llevar de nuevo al robot al comienzo con las flechas naranjas(), y después, cambiar el programa para dar las instrucciones correctas.

Lightbot va dando más instrucciones a medida que completa más niveles.

No se desanime si su programa no es correcto en su primer intento. Basta leer cada instrucción que le dio Lightbot cuidadosamente e imaginar cómo sería y debe actuar en cada paso.

Si te encuentras atascado, pregunta a otros compañeros o al instructor en busca de ayuda. Explique su programa paso a paso a otra persona, y ver si se puede detectar un error. En el mundo real de la Programación es un esfuerzo de colaboración - es normal e buscar consejos al equipo para poder conseguir la mejor solución para resolver los puzzles.

Posibles soluciones a pantallas de lightboot free.

No siempre un puzzle se resuelve de la misma manera, es cierto que hay resultados más óptimos que otros.

Básicos

Últimos Tutoriales del Autor

» Taller de Ventas

» SCRATCH, aprende a programar desde niño

» Contando mi experiencia en los talleres gratuitos de Apple



1.



Se necesitan 3 comandos.

Se tiene que pulsar cualquier tecla o en cualquier lugar para continuar.

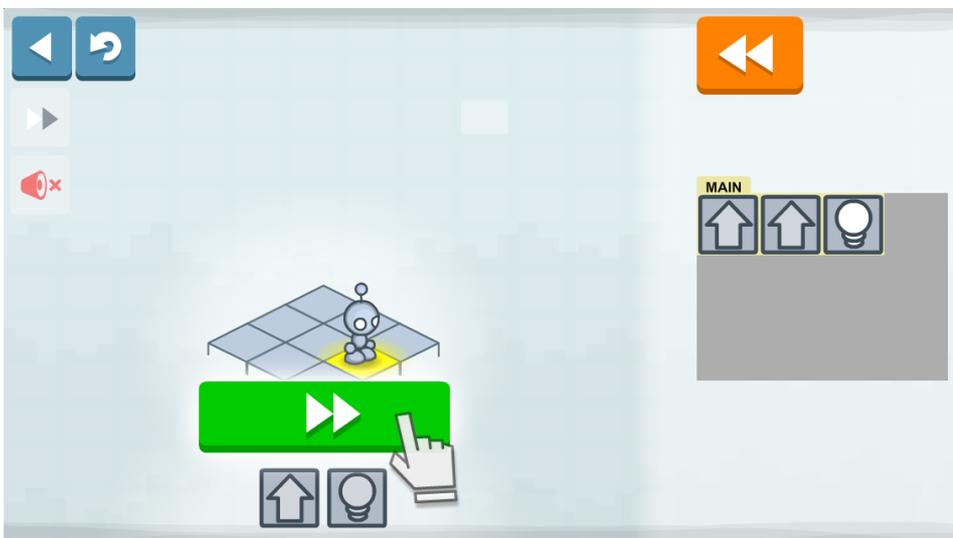
El robot necesita ayuda para iluminar todos los cuadros azules.

El comando Caminar, dice que camine hacia delante.

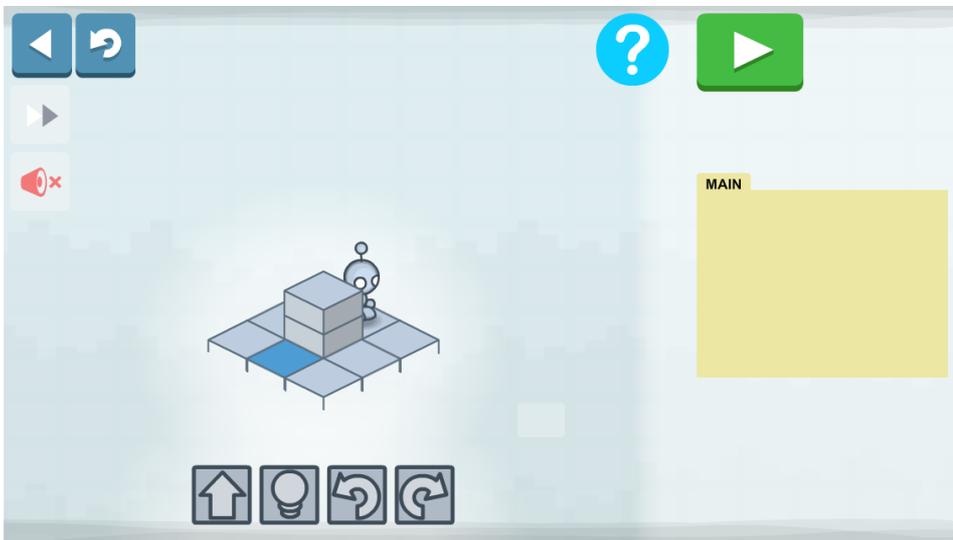
El comando iluminar hace que se ilumine el cuadro donde estoy situado.

Para darme las ordenes hay que pulsar los comandos que se sitúan abajo y se añadirán al área Main.

Cuando se haya terminado se presiona el triángulo verde de play.



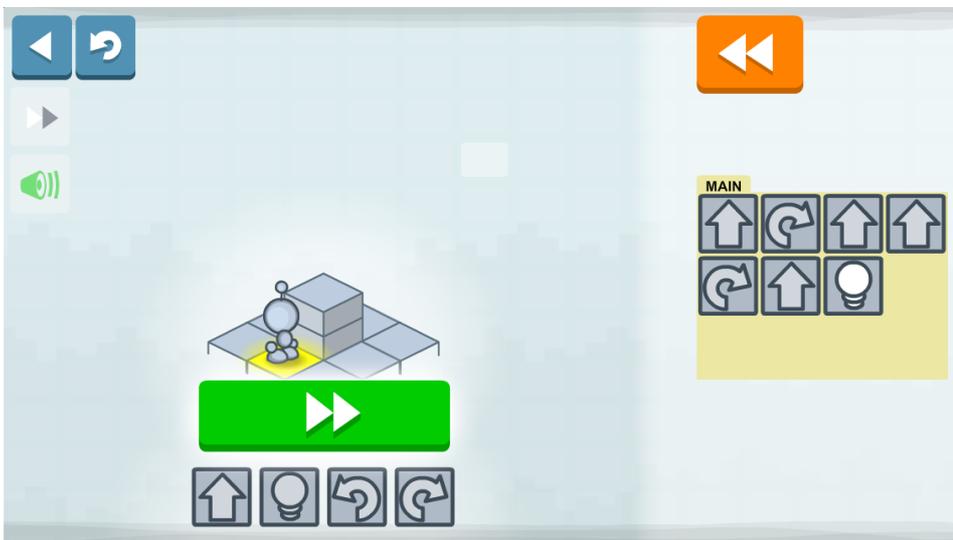
2.



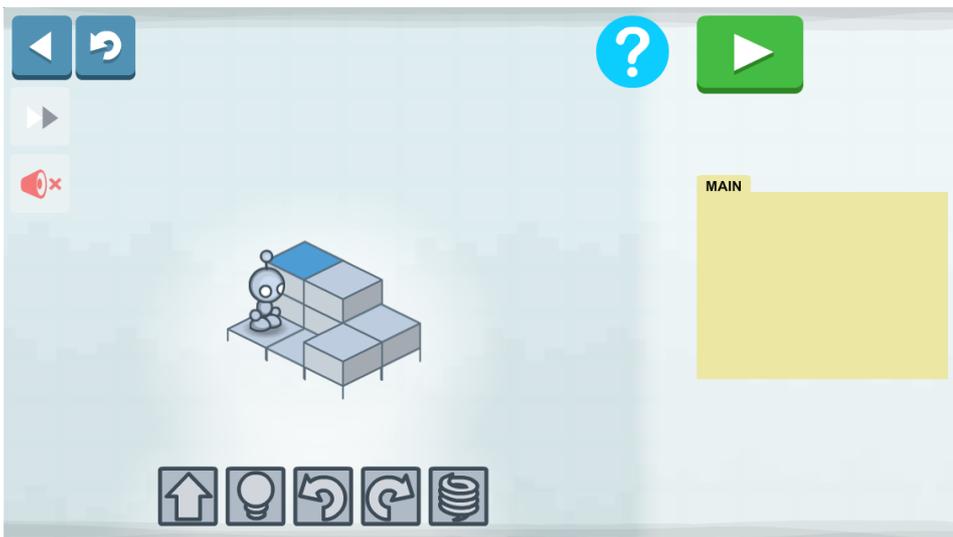
Se necesitan 7 comandos.

Se incorporan nuevos comandos : Izquierda me dice que gire a la izquierda (en contra de las agujas del reloj)

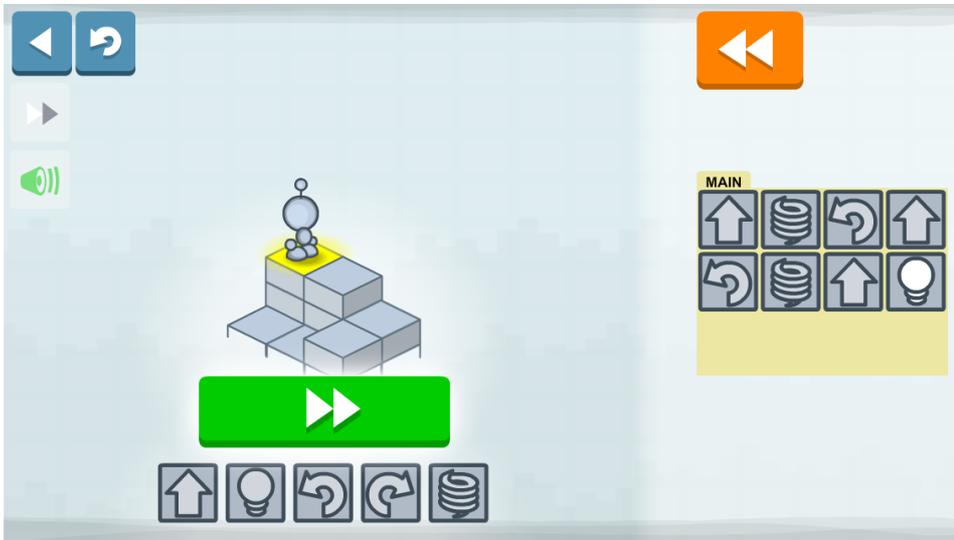
Derecha (igual que las manecillas del reloj)



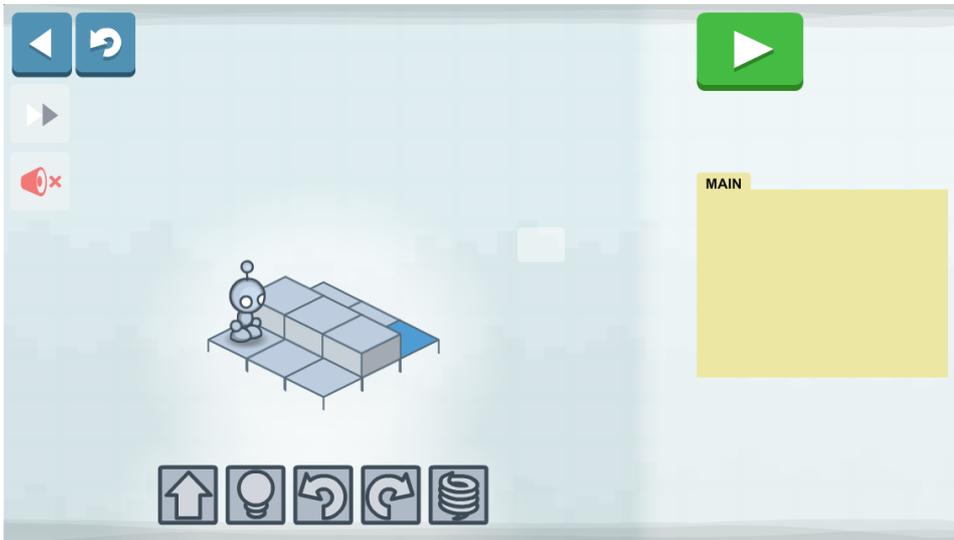
3.



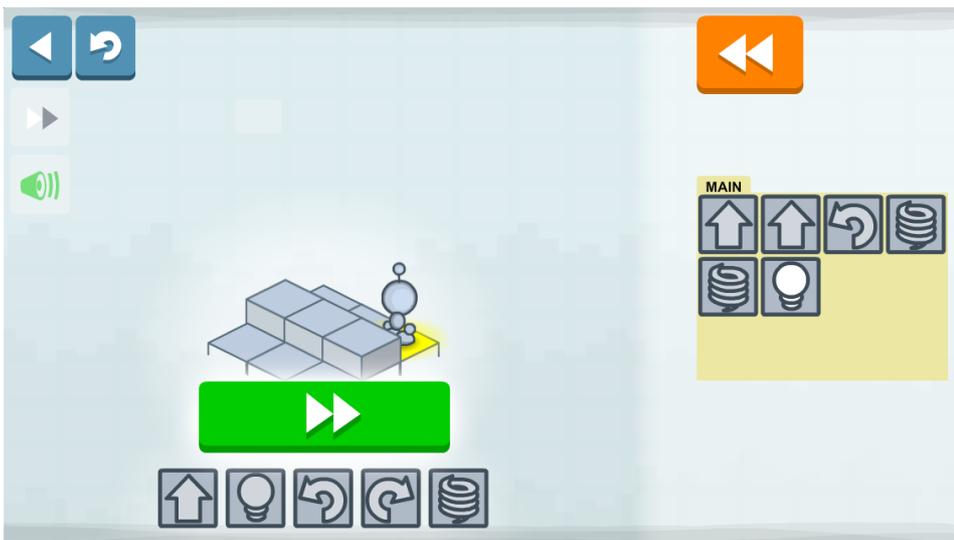
Se necesitan 8 comandos, se incluye el comando salto, me dice que Salte, bien hacia arriba o hacia abajo.



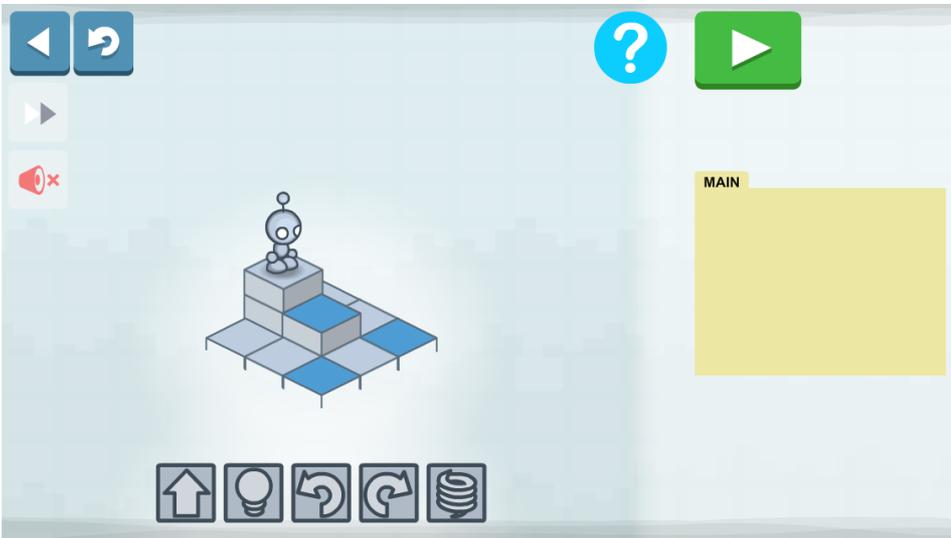
4.



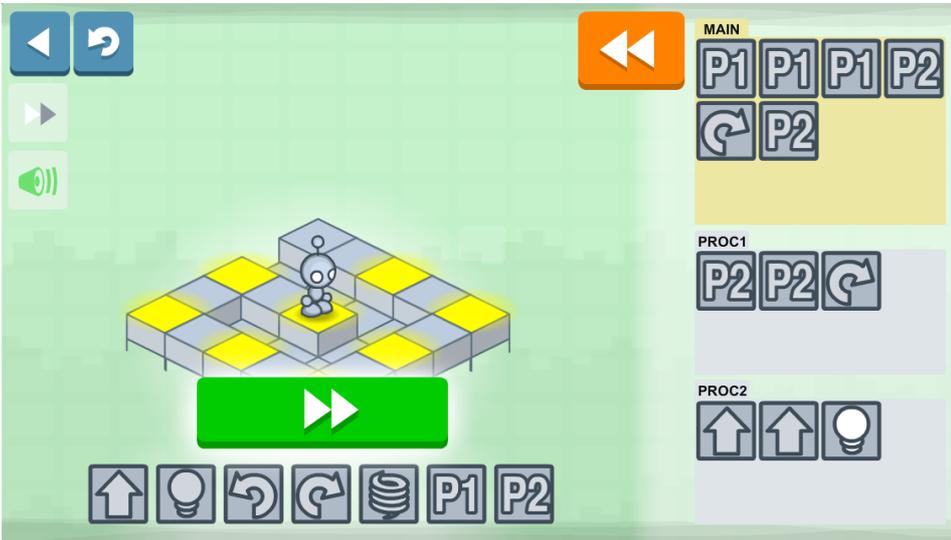
Se necesitan 6 comandos



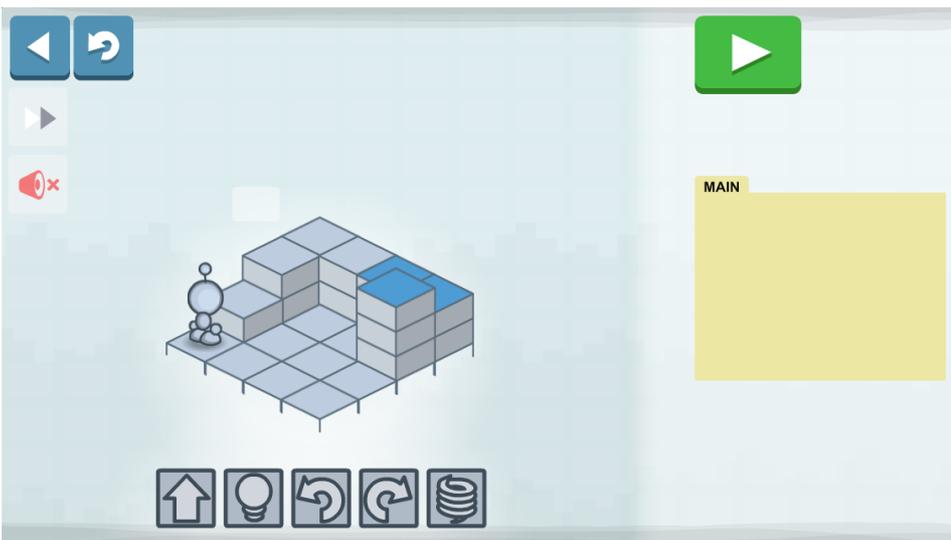
5.



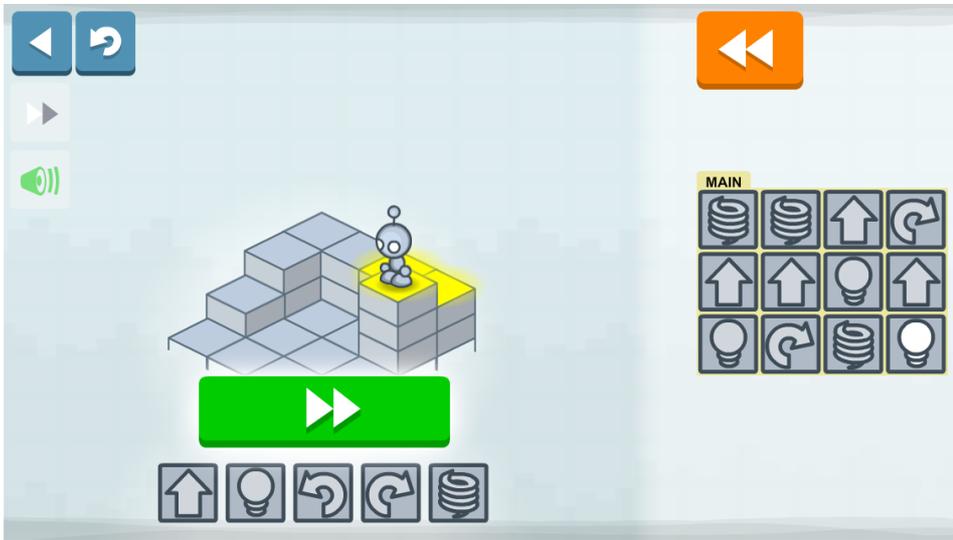
Se necesitan 11 comandos. Recuerda que hay que iluminar todos los cuadros azules.



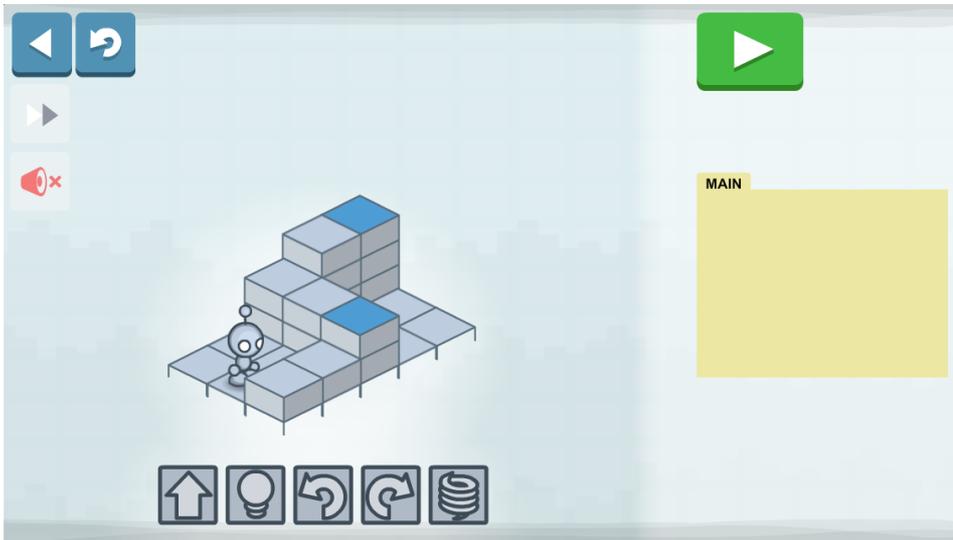
6.



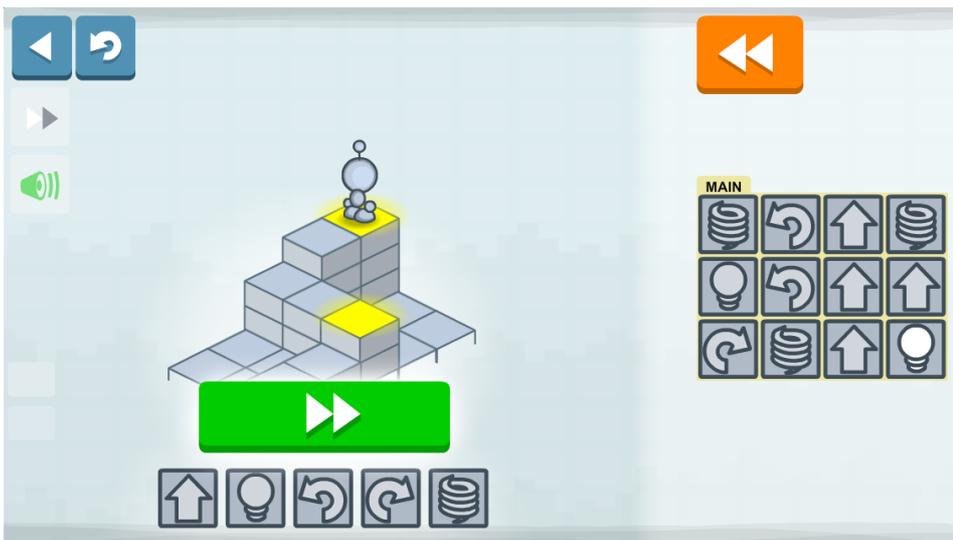
se necesitan 12 comandos



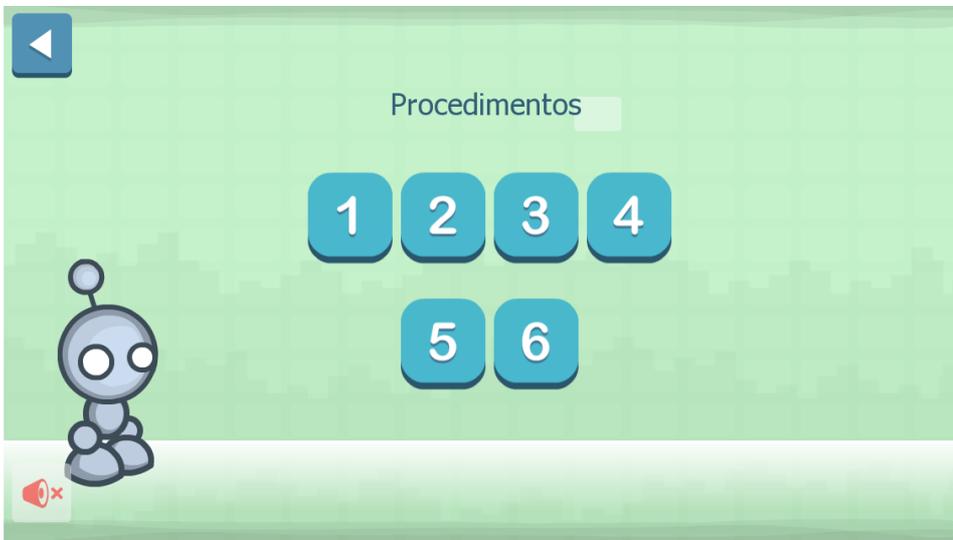
7.



Se necesitan 12 comandos



Procedimientos

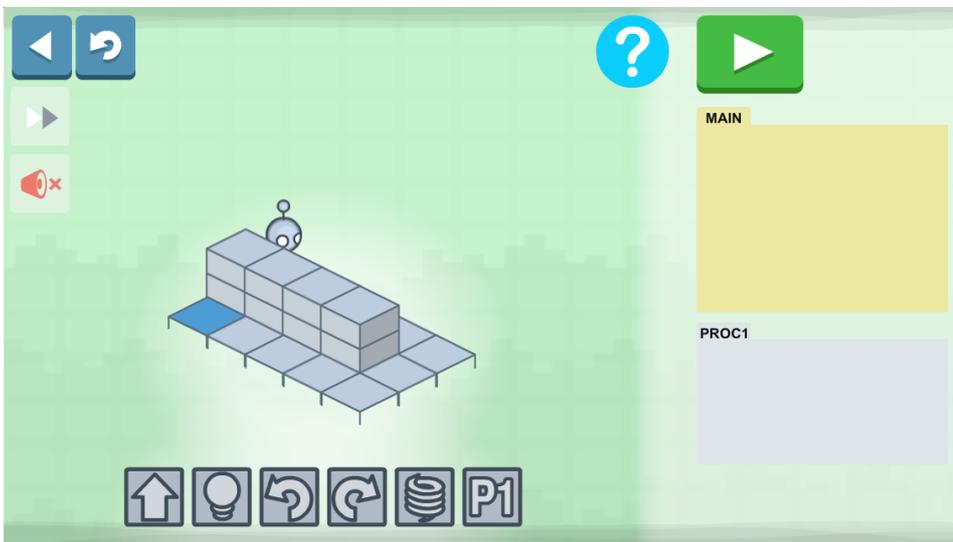


1.

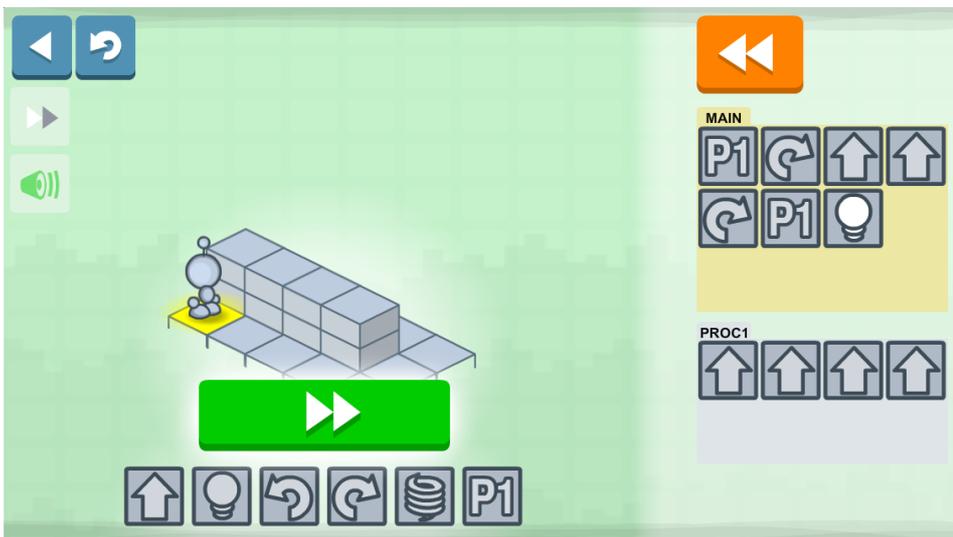
A partir de ahora necesitaras más espacio que el área MAIN para los próximos desafíos.

Se podrán añadir más comandos a mi nueva área PROC1.

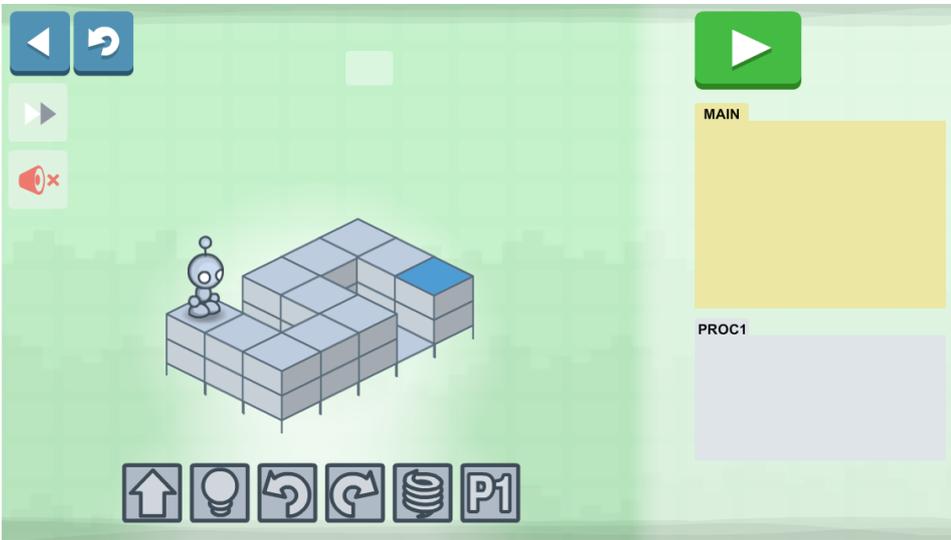
El nuevo comando P1 me dice que ejecute los comandos que hay en PROC1.



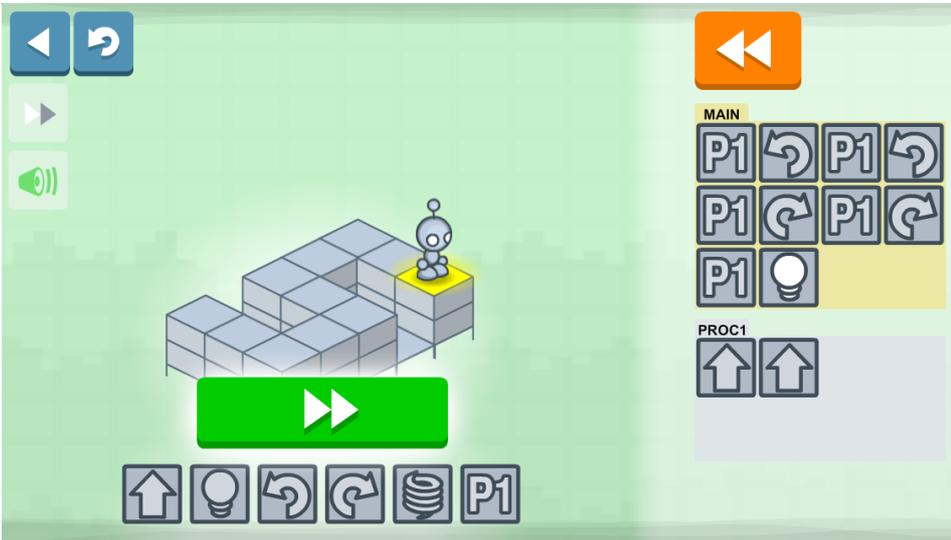
Para este reto se necesitaran 11 comandos.



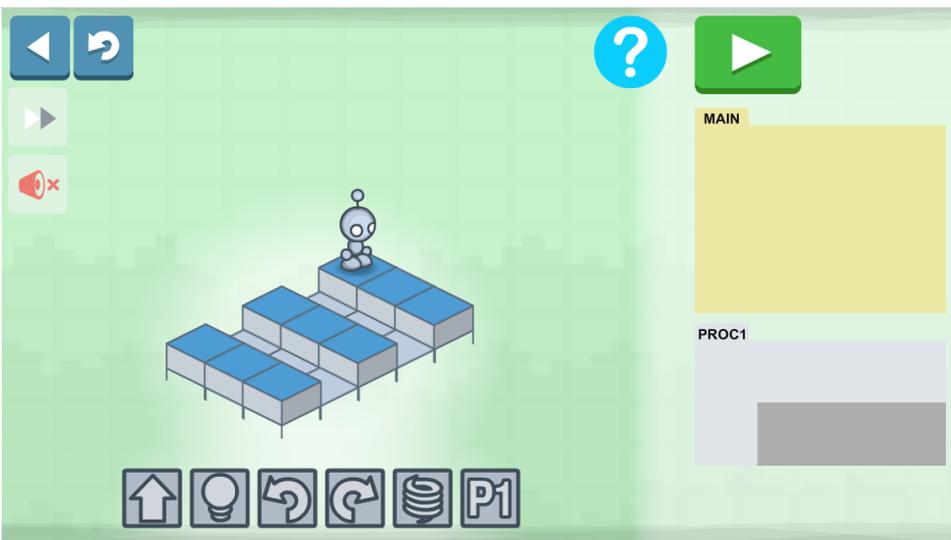
2.



Se necesitan 12 comandos.



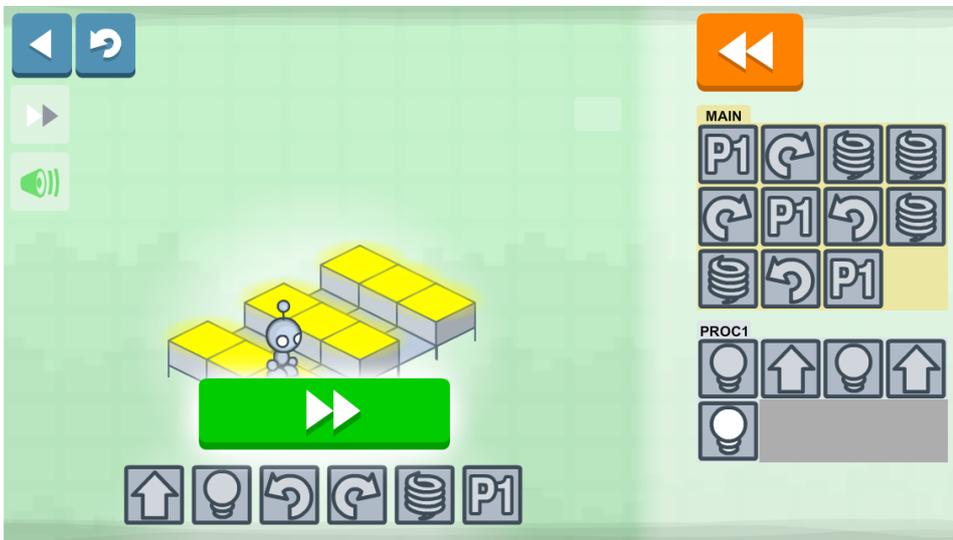
3.



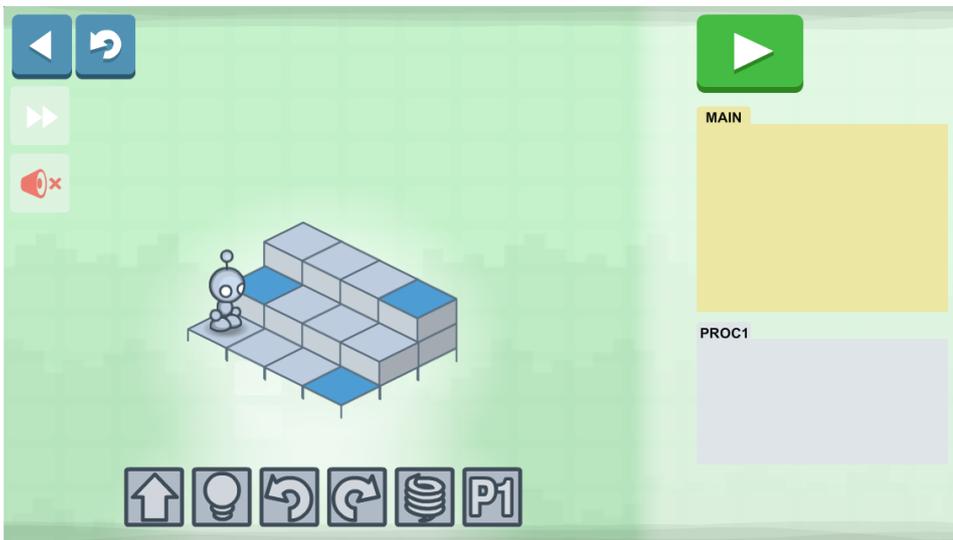
El área "PROC1" es muy útil para los patrones de comandos.

Puedes re-usar parones en PROC1 usando varios comandos P1

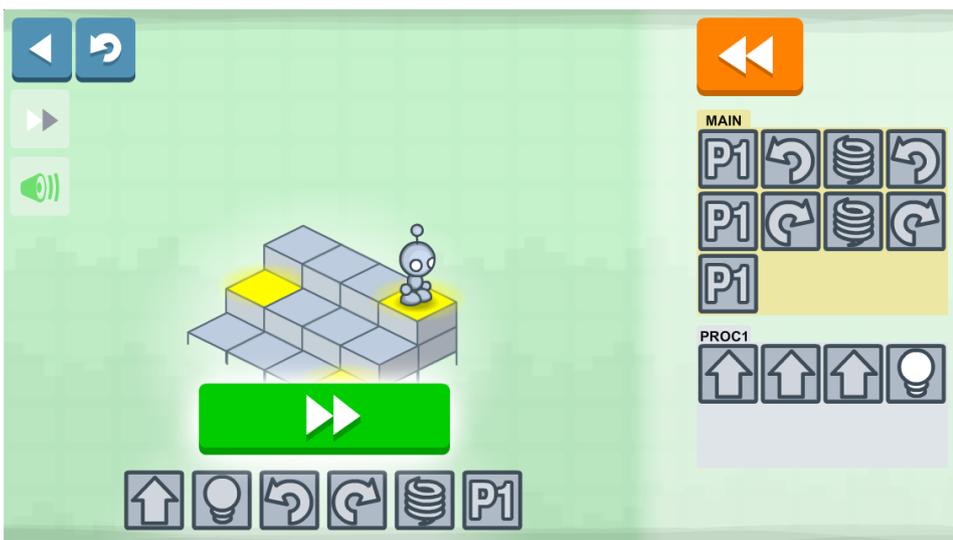
Para este reto se necesitan 16 comandos.



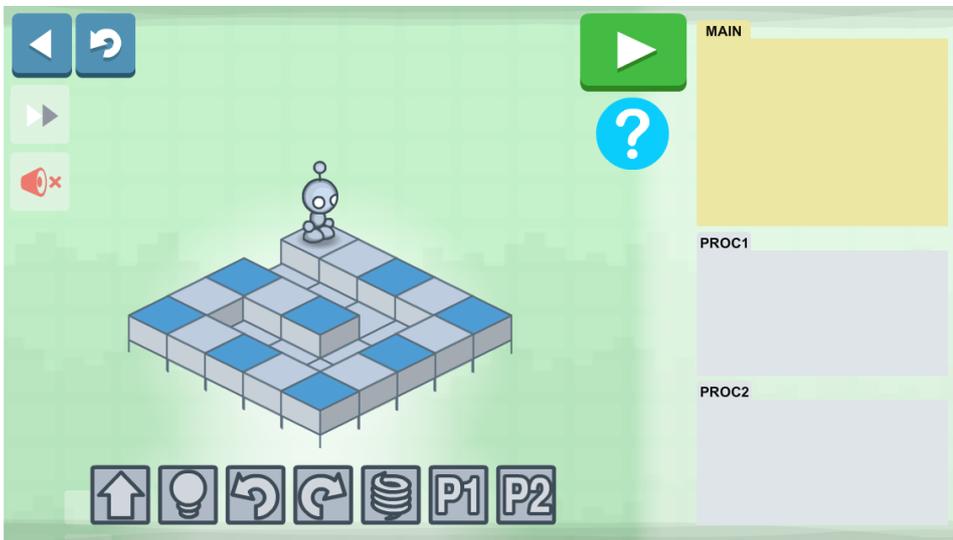
4.



Se necesitarán 13 comandos.

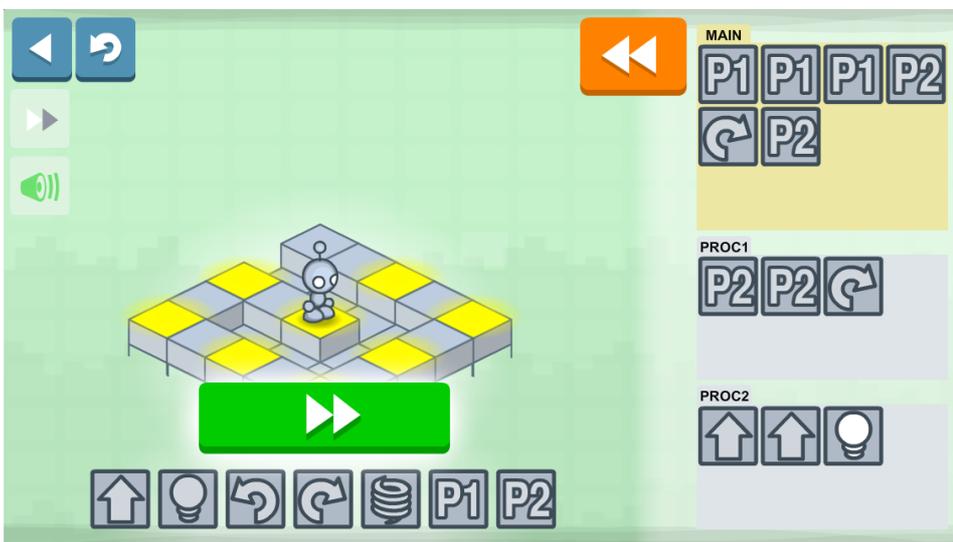


5.

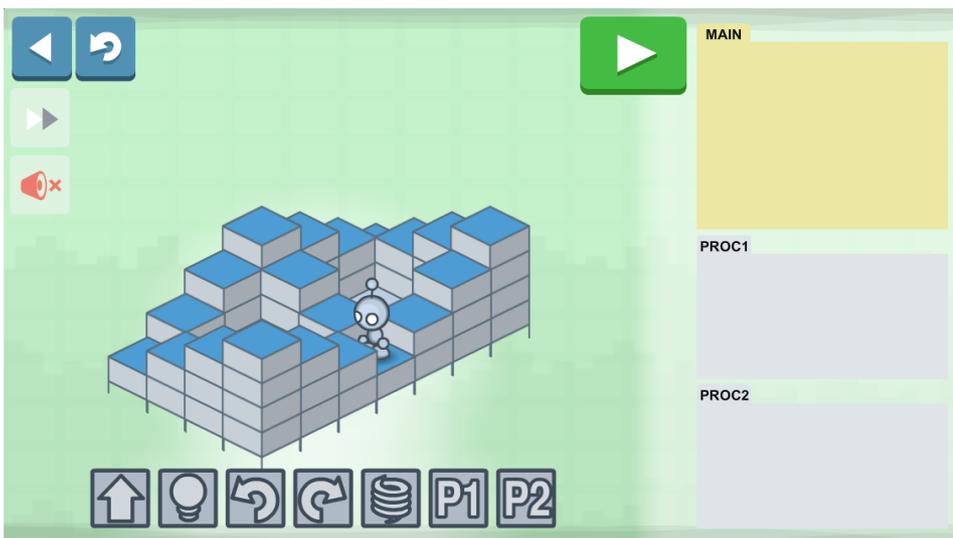


Se necesitan 12 comandos.

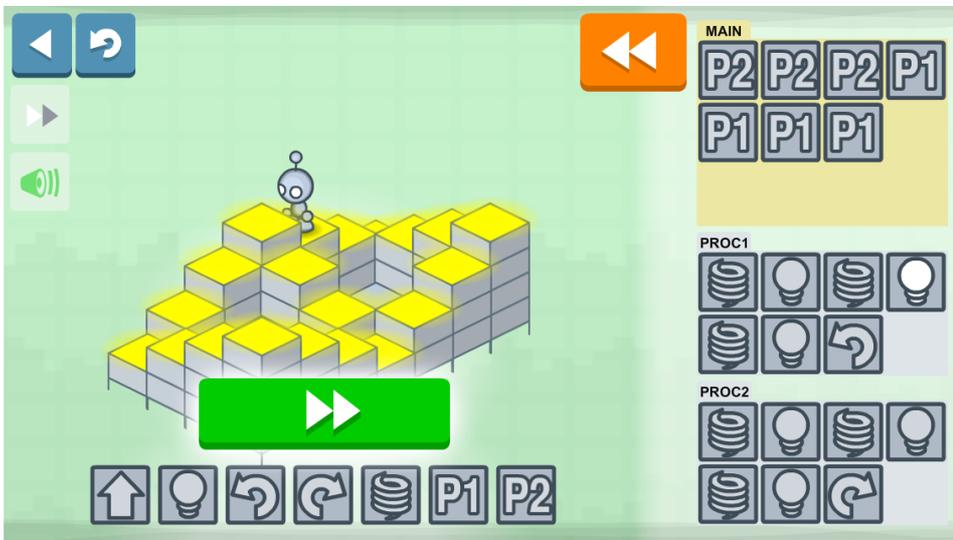
Utilizaremos otra área más "PROC2", "P2", ejecutara los comandos que haya en "PROC2". Se puede usar "P2" en "PROC1".



6.



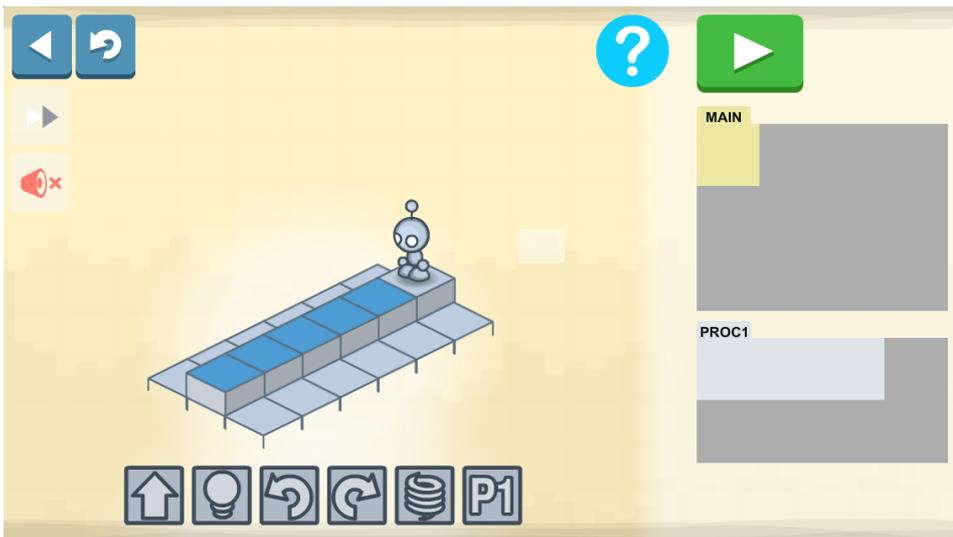
Se necesitan 21 comandos.



Bucles



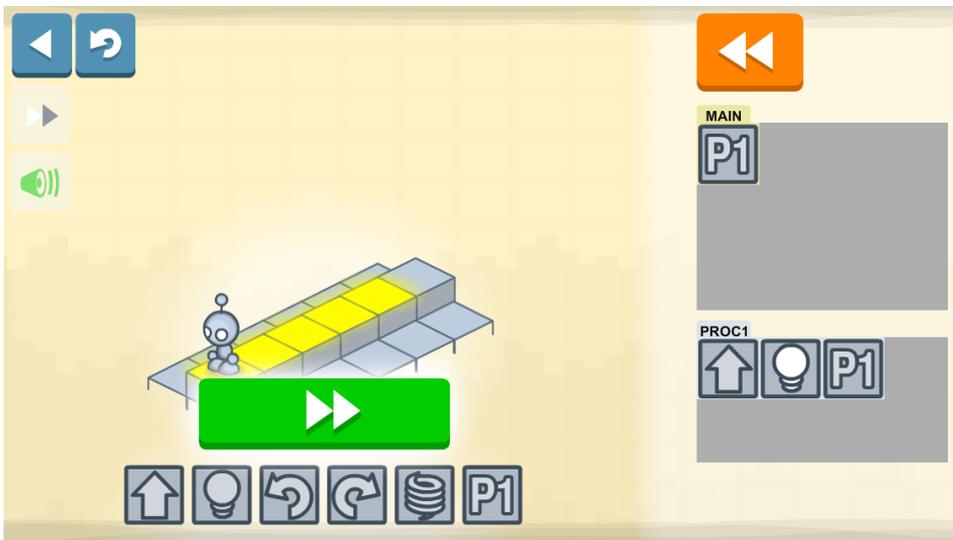
1.



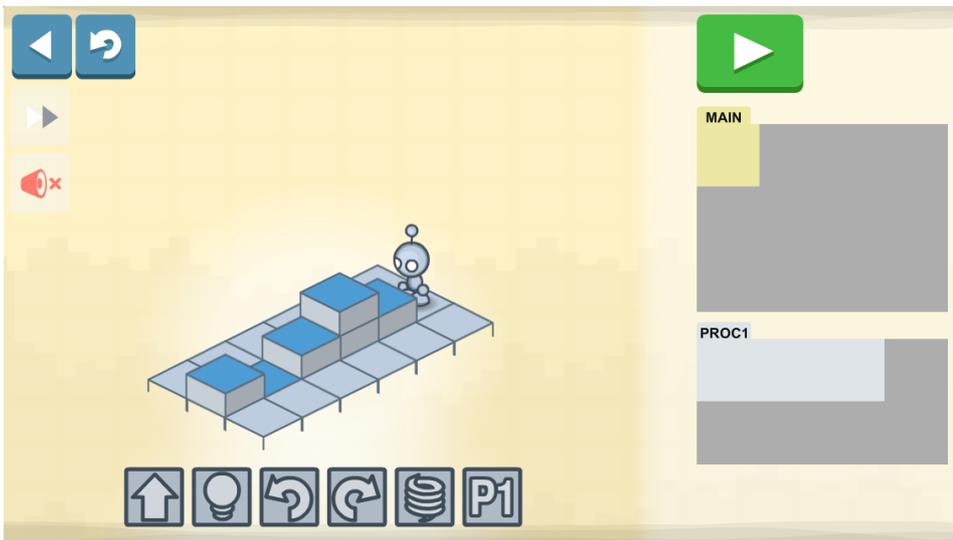
Se puede usar el comando "P1" dentro de "PROC1" para crear un bucle.

Cuando uses "P1" dentro de "PROC1", se volverá a ejecutar "PROC1" una y otra vez.

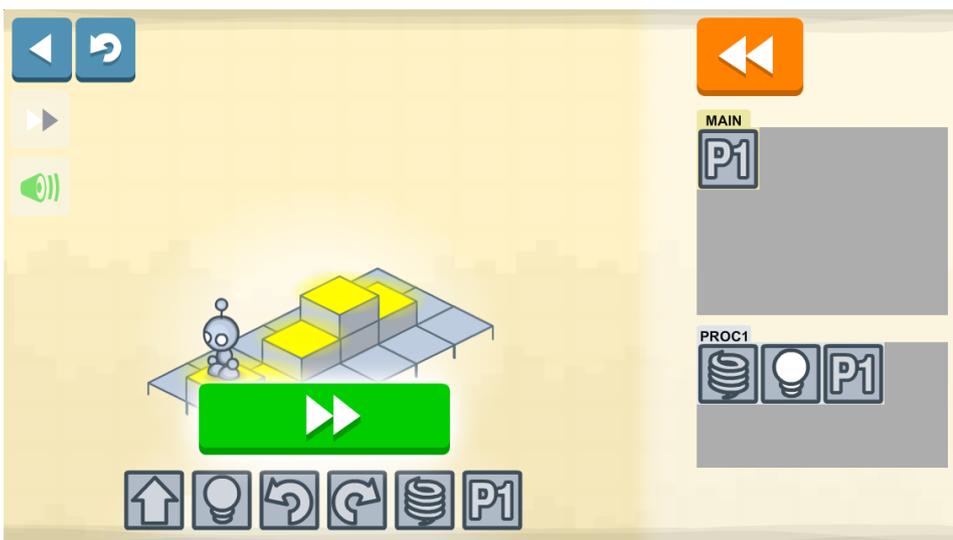
Necesitarás cuatro comandos para completar el reto.



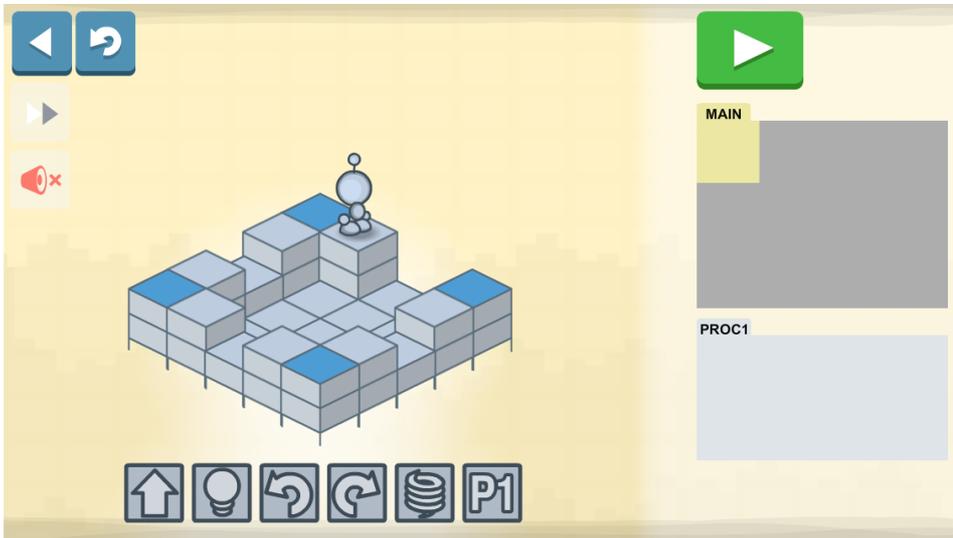
2.



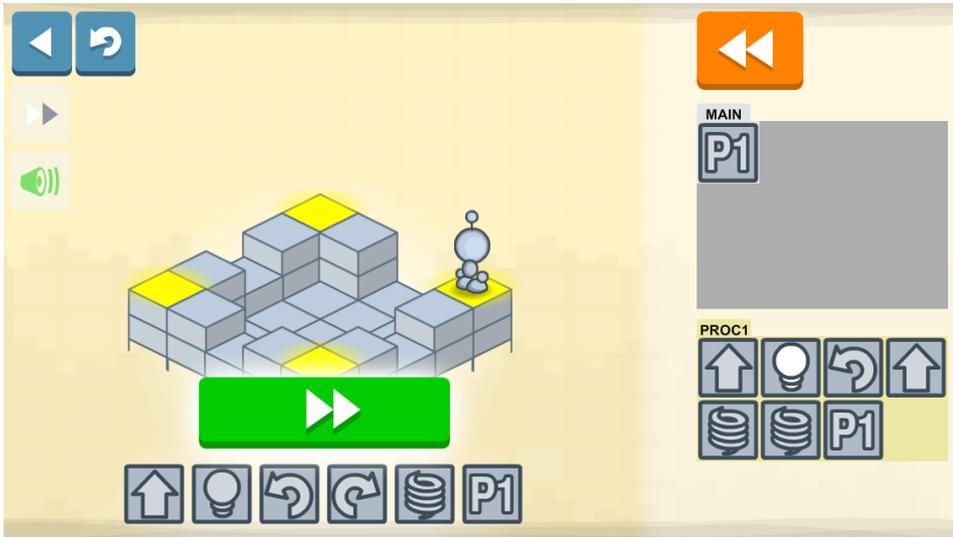
Se necesitarán cuatro comandos.



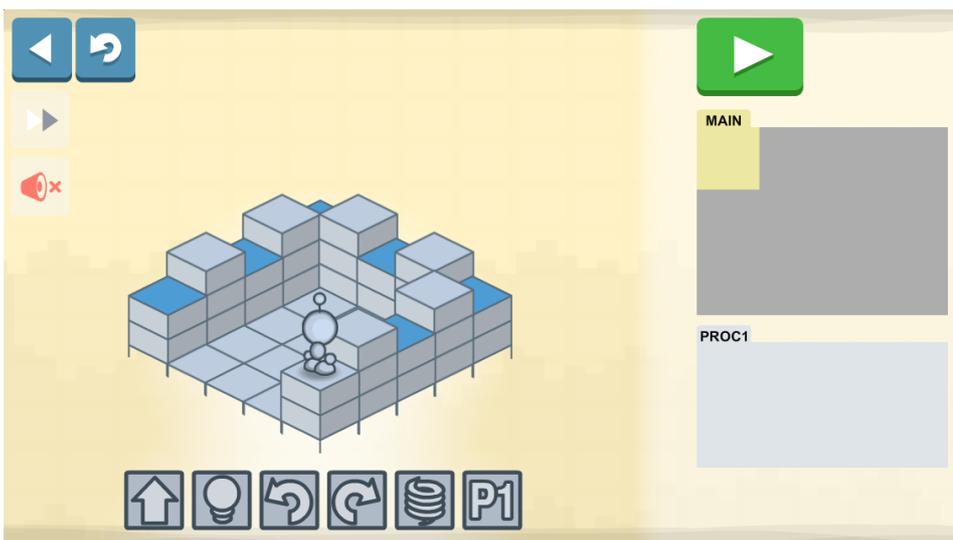
3.



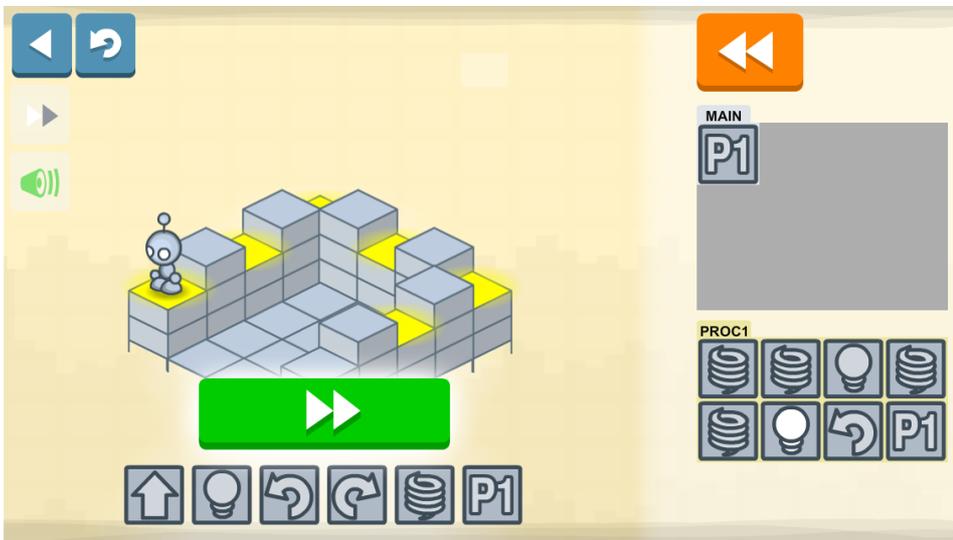
Se necesitan ocho comandos.



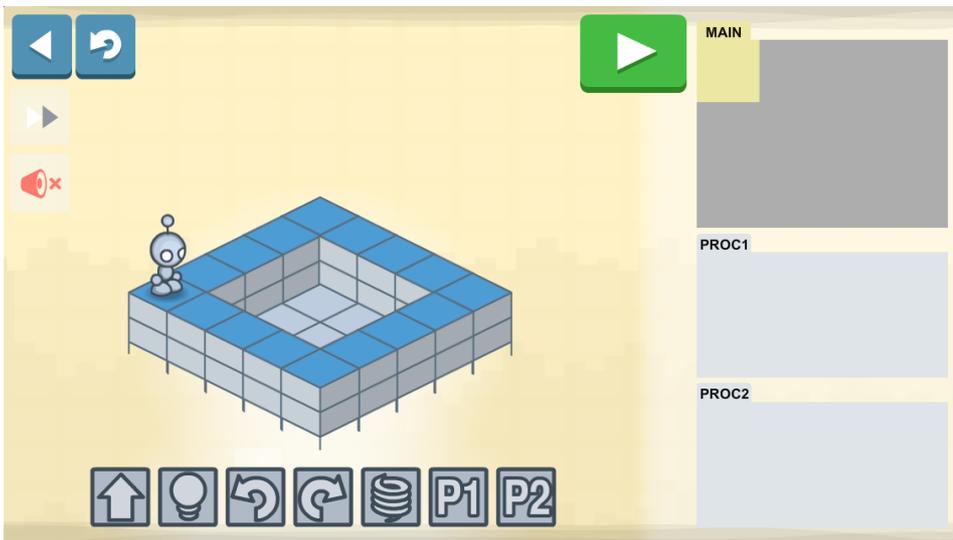
4.



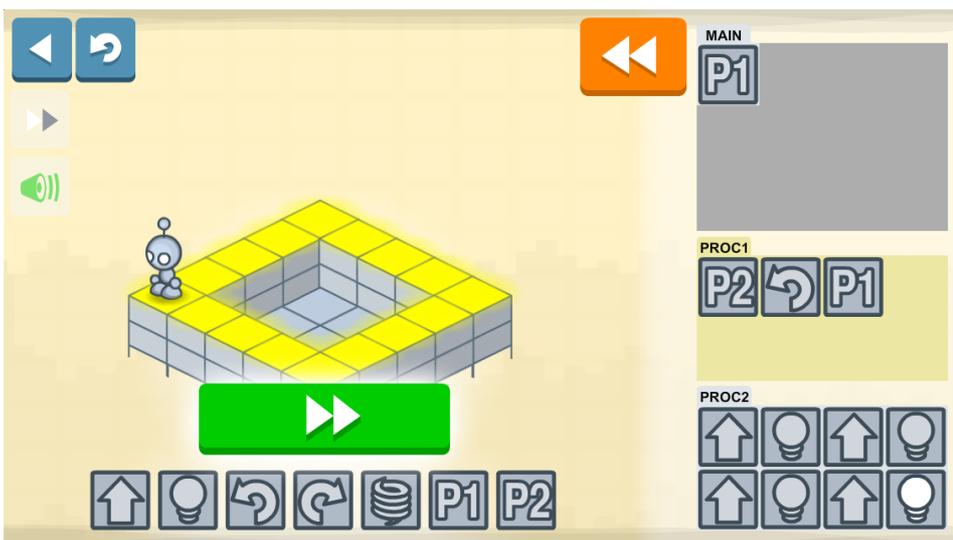
Se necesitan nueve comandos.



5.



Se necesitan 12 comandos



Si os ha gustado superar estos niveles de la versión gratuita tenéis que saber que existen dos versiones más de cuatro a ocho años y otra de más de nueve.

Espero que lo disfrutéis.

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

Share |

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

» [Regístrate](#) y accede a esta y otras ventajas «



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

IMPULSA

Impulsores

Comunidad

¿Ayuda?

sin clicks

0 personas han traído clicks a esta página

+ + + + + + + +

powered by [kamacracy](#)

Copyright 2003-2014 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

