

FICHA¹

DATOS, ALTERNATIVA CONDICIONAL Y FUNCIONES

Guía para docentes y familias



¹Material extraído del Manual para la Enseñanza de las Ciencias de la Computación en el aula de la Iniciativa Program.AR. Martínez López, Pablo E., Klinkovich, V., Czemerinski, H., Miller, Ignacio D., ... Andújar, L. (2019). Ciencias de la computación para el aula, 1er ciclo secundaria (1st ed.). Buenos Aires, Argentina: Fundación Sadosky.

NOMBRE Y APELLIDO:

CURSO:

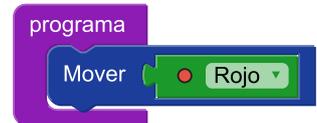
FECHA:

TIPOS DE DATOS



¿Los colores y las direcciones son lo mismo?
¿Por qué a veces usamos unos y, otras veces, otros?

1. ¿Qué ocurre si le damos la instrucción **Poner [Sur]** al cabezal de Gobstones?
¿Y si le damos la instrucción **Mover [Rojo]**? Justificá tu respuesta.



Ahora, abrí el entorno de Gobstones, creá un nuevo proyecto, armá ambos programas y fijate qué sucede. ¿Es lo que vos esperabas?

2. Este es un programa para dibujar un cuadrado de 4×4 con bolitas verdes. Sin embargo, no funciona bien. Miralo con atención y marcá todos los errores que encuentres.

Ahora, abrí el proyecto "Por qué falla", corregí todos los errores que identificaste y ejecutalo. ¿Te faltó encontrar alguno? Realizá todos los cambios que hagan falta para que el programa dibuje el cuadrado.

TIPOS DE DATOS

Los tipos de datos son conjuntos de valores que tienen características comunes. En Gobstones tenemos los siguientes: *colores*, *direcciones*, *números* y *booleanos*. Ya veremos de qué se tratan estos últimos.



NOMBRE Y APELLIDO:

CURSO:

FECHA:

EL CABEZAL PUEDE CONTAR

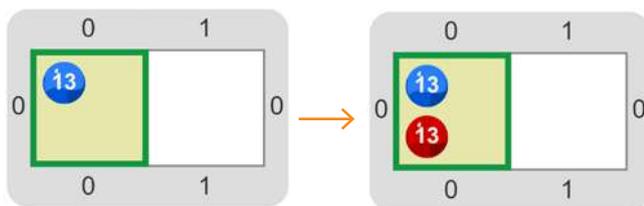
Los mimos son actores que se valen exclusivamente de gestos y de movimientos corporales para actuar ante el público. Algunos se especializan en imitar a otras personas que ven a su alrededor. ¡Ahora vamos a hacer que las bolitas rojas hagan de mimos imitando a las de las de otros colores!



1. Abrió el proyecto "El cabezal juega a ser mimo" y construí un programa que ponga bolitas rojas para igualar la cantidad de bolitas azules.

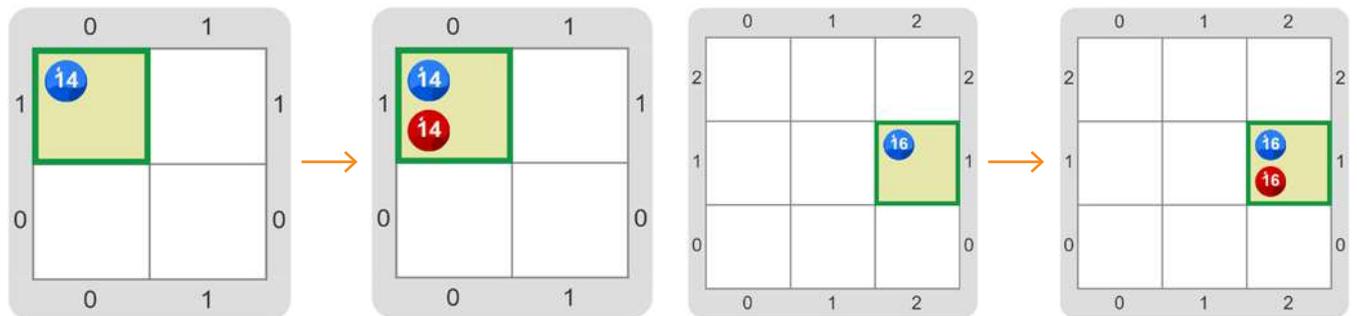
¡ATENCIÓN!

La cantidad de bolitas azules va variando de ejecución en ejecución. Te mostramos algunos tableros iniciales y sus correspondientes tableros finales.



Tablero inicial

Tablero final



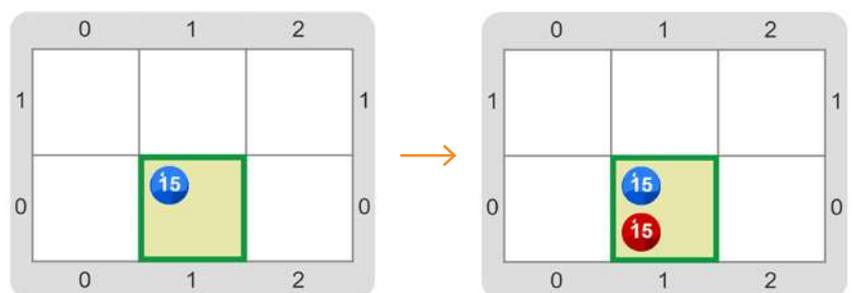
Tablero inicial

Tablero final

Tablero inicial

Tablero final

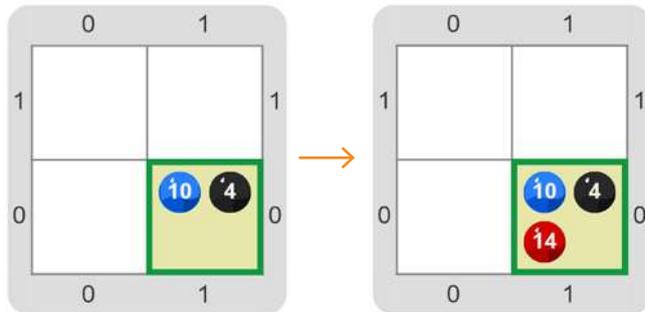
Recordá que el mismo programa tiene que servir para transformar cualquiera de los tableros iniciales. Probalo cuantas veces necesites.



Tablero inicial

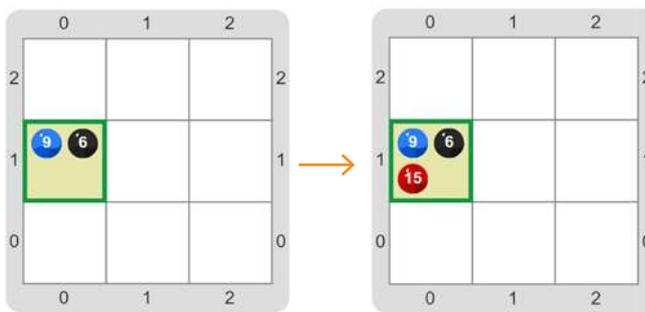
Tablero final

2. Ahora abrí el proyecto "El mimo que suma" y hacé un programa que ponga tantas bolitas rojas como haya de azules y negras. El desafío es que en tu programa uses una única repetición. Buscá una forma de que el número de bolitas azules y el número de bolitas negras se puedan usar juntos en la repetición. Te mostramos algunos tableros iniciales y finales:



Tablero inicial

Tablero final



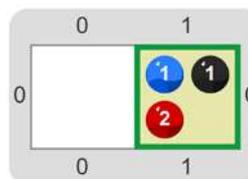
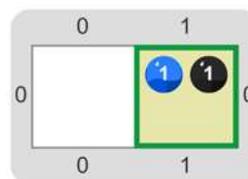
Tablero inicial

Tablero final

Explorá el entorno para ver si hay alguna herramienta nueva que te ayude a resolver el desafío.

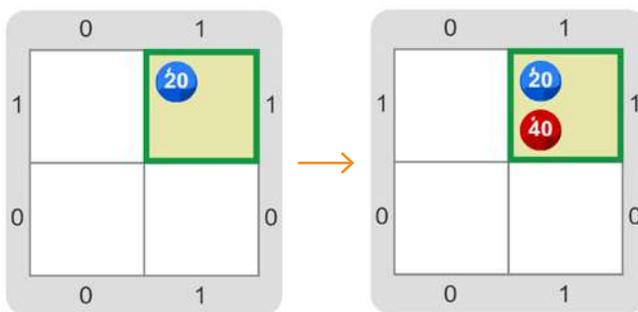


Tablero inicial



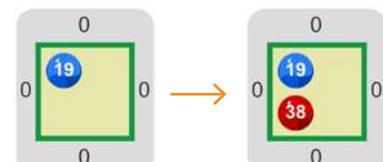
Tablero final

3. Abrí el proyecto "Supermimo multiplica" y completá el programa. Esta vez hay que poner el doble de bolitas rojas que de azules. El desafío, nuevamente, es hacerlo con una única repetición. Además, la solución debe usar una sola vez el comando `Poner []` y una sola vez el sensor `número de bolitas []`. Mirá algunas de las combinaciones de tablero inicial y final.



Tablero inicial

Tablero final



Tablero inicial

Tablero final



Tablero inicial

Tablero final

4. ¿Qué bloques nuevos usaste para resolver los desafíos?

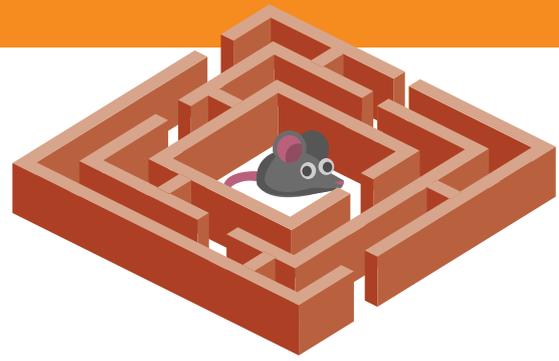
NOMBRE Y APELLIDO:

CURSO:

FECHA:

LABERINTO

¡Un ratoncito está atrapado en un laberinto y necesita de tu ayuda para poder escapar! Por suerte, hay flechas que marcan el camino hacia la salida, que se encuentra a solo 20 pasos.



1. Abrí el proyecto "Laberinto", y completá los procedimientos **Salir del laberinto** y **Avanzar un paso siguiendo la flecha** de modo que le indiquen al ratón lo que debe hacer para escapar.

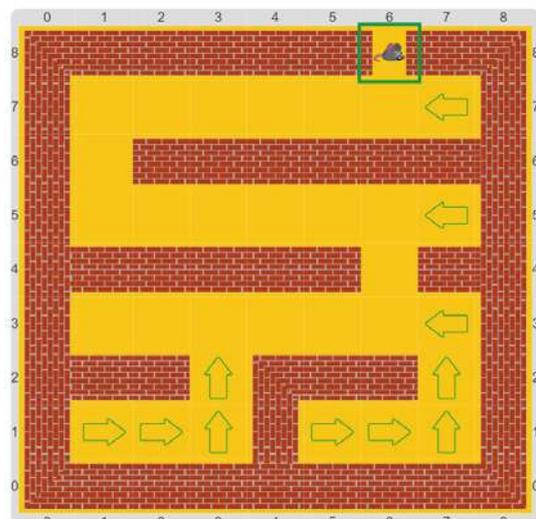
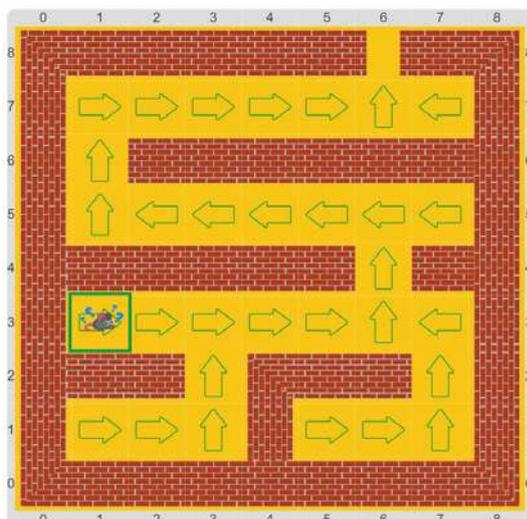
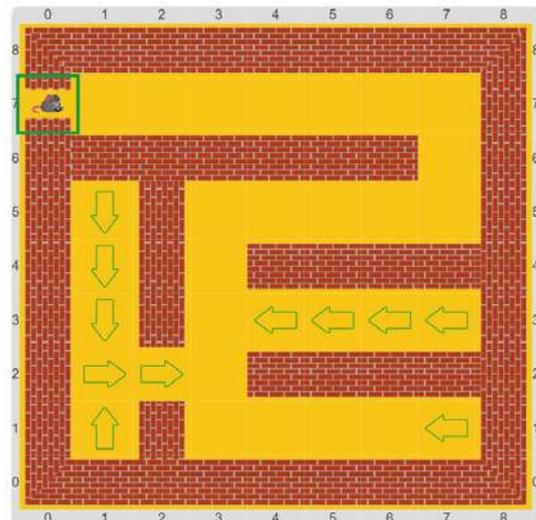
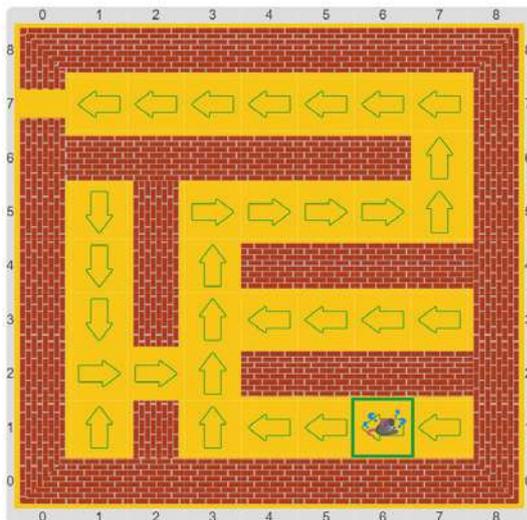
Definir **Salir del laberinto**



Definir **Avanzar un paso siguiendo la flecha**



Te mostramos algunos laberintos y cómo tienen que quedar al terminar:



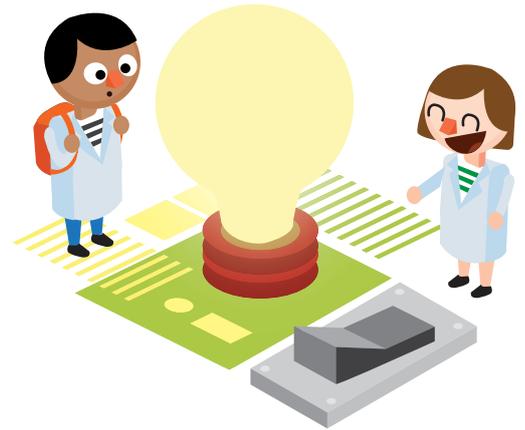
NOMBRE Y APELLIDO:

CURSO:

FECHA:

APAGAR LA LUZ O PRENDERLA

¿Cómo funciona un interruptor de luz? Al accionarlo, cambia el estado de una lamparita: si está apagada, la prende; si está prendida, la apaga.



1. Abrí el proyecto "Apagar la luz o prenderla", y completé el procedimiento **simular interruptor de luz**. Tené en cuenta que, *a priori*, no sabemos si la lamparita está encendida o apagada. ¡El programa tiene que funcionar en ambos casos!

Definir **Simular interruptor de luz**

COMPLETAR

¡A TENER EN CUENTA LA VESTIMENTA! 👁

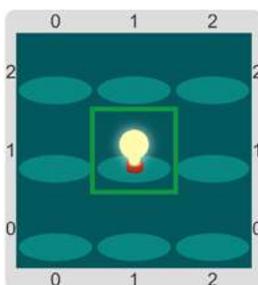
No hay procedimientos en la biblioteca. Para resolver el desafío, hay que considerar la vestimenta de la actividad.

● → lamparita prendida

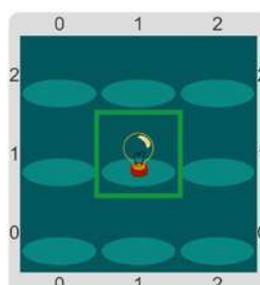
● → lamparita apagada



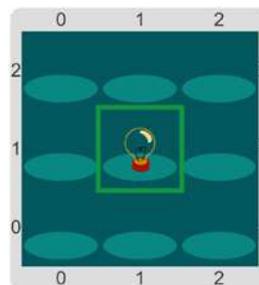
Los posibles tableros iniciales y sus correspondientes tableros finales son los siguientes:



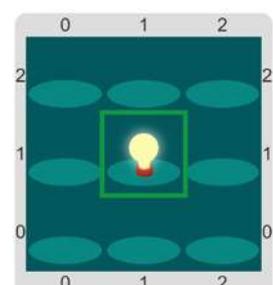
Tablero inicial



Tablero final



Tablero inicial



Tablero final

Copíá acá abajo cómo quedó tu programa.

¿APAGAR O PRENDER?

¿Cómo sabemos si tenemos que prender o apagar la luz? Explorá el entorno para encontrar las herramientas que hacen falta. ¡Acordate de usar procedimientos!



NOMBRE Y APELLIDO:

CURSO:

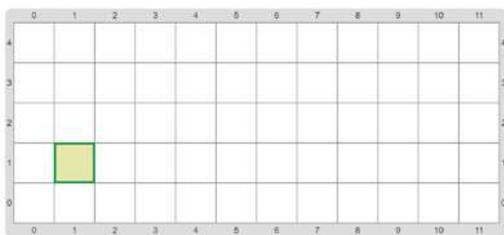
FECHA:

NO ME QUIERO CAER

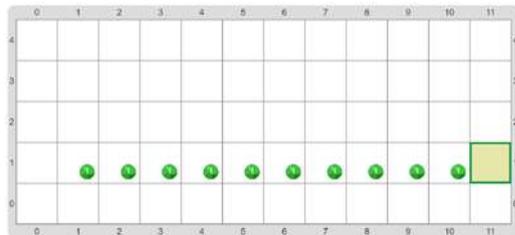
¡Las víboras vuelven más verdes que nunca!

1. Abrió el proyecto "No me quiero caer" y construí un programa para dibujar una víbora con 10 bolitas verdes. Tenés que colocar la primera bolita en la celda donde inicialmente se encuentra el cabezal y continuarla hacia el este. Si hay suficiente lugar, la víbora queda alineada; pero si en algún momento no puede moverse más hacia el este, se acaba el tablero, dobla y sigue hacia el norte.

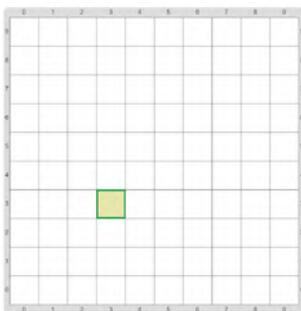
Las siguientes figuras muestran algunos tableros iniciales y los tableros finales correspondientes:



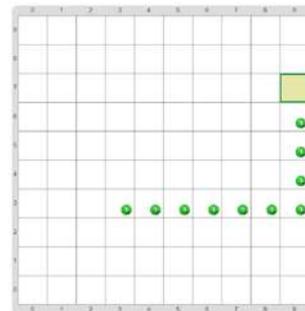
Tablero inicial (11 lugares al este)



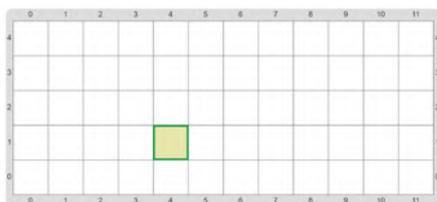
Tablero final



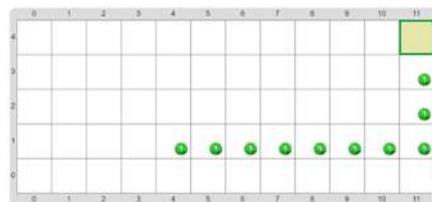
Tablero inicial (7 lugares al este)



Tablero final



Tablero inicial (8 lugares al este)



Tablero final

¿CÓMO EVITAR QUE EL CABEZAL HAGA BOOM?

Explorá el entorno en busca de algo que te permita determinar si el cabezal puede moverse en una dirección específica.

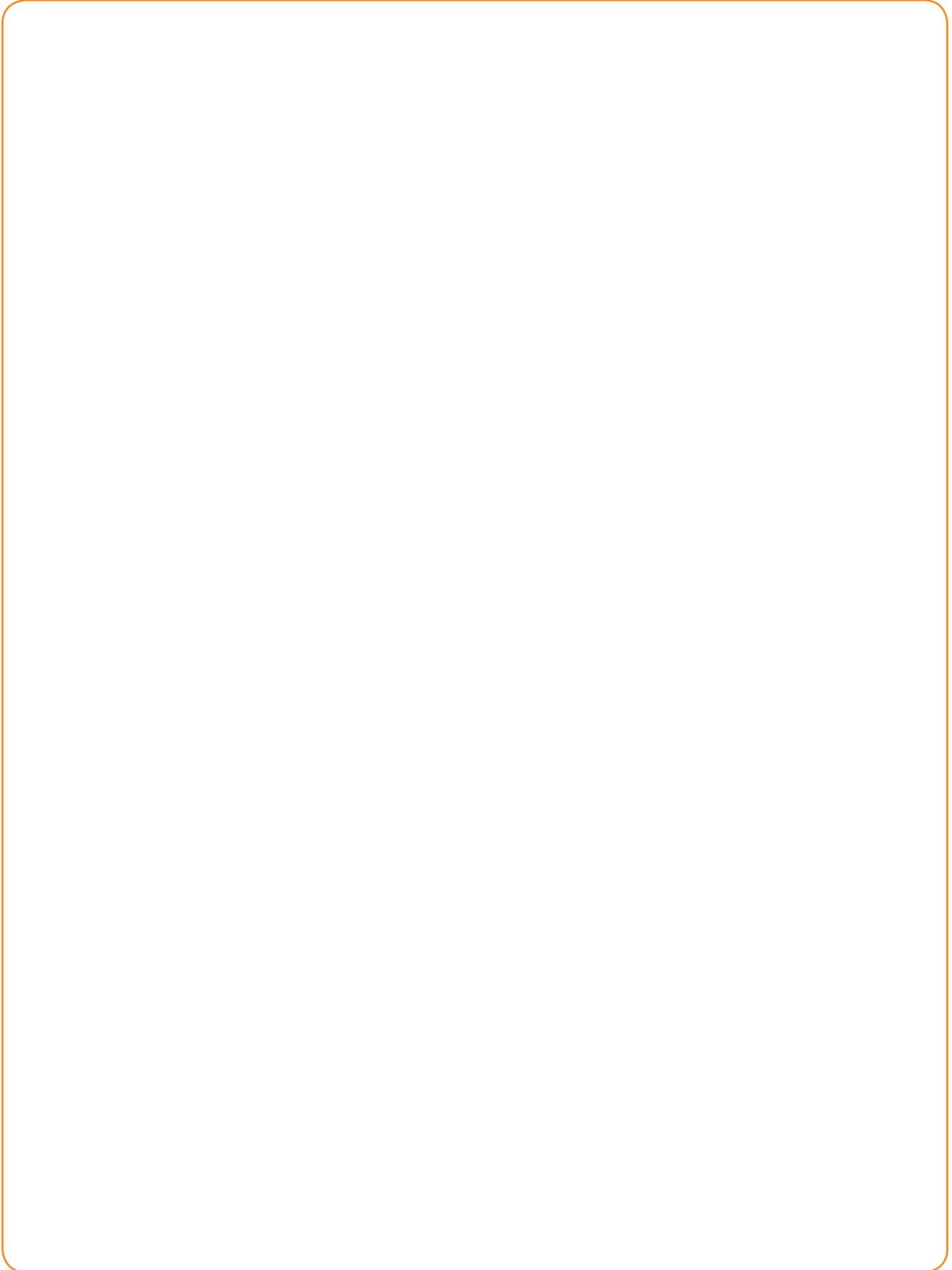


NOMBRE Y APELLIDO:

CURSO:

FECHA:

2. Copiá acá abajo cómo quedó tu programa.

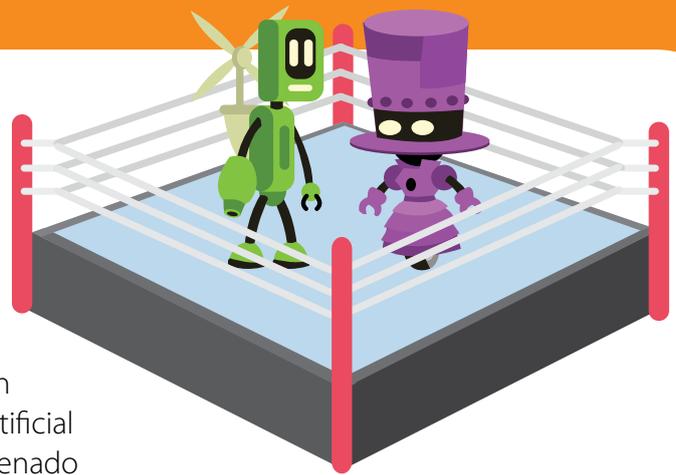


NOMBRE Y APELLIDO:

CURSO:

FECHA:

QUE GANE EL MEJOR



Verdolaga y Copa Alta son dos robots que fueron entrenados mediante técnicas de inteligencia artificial para combatir. Cuando se enfrentan, el más entrenado gana. En esta actividad, tenés que construir un programa para simular un combate entre Verdolaga y Copa Alta.

1. Abrí el proyecto "Que gane el mejor" y construí un programa que retire del tablero al robot vencido en combate. Si los dos tienen el mismo nivel de entrenamiento, se produce un empate y ambos deben retirarse.

EL INICIAL DEL FINAL

Luego de ejecutar un programa, podés volver a ver el tablero inicial haciendo clic en Tablero inicial.

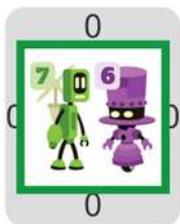


ALGUNAS AYUDITAS

- Activá y desactivá la vestimenta para ver cómo están representados los robots y sus niveles de entrenamiento.
- Buscá en el entorno algunos bloques que te permitan comparar el nivel de entrenamiento de los robots.
- ¡Inspeccioná la biblioteca! A lo mejor encontrarás algo que te ayuda a construir tu programa.



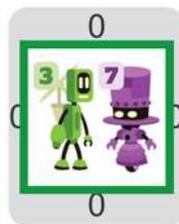
Acá podés ver algunos tableros iniciales y cómo deben quedar al finalizar la ejecución del programa.



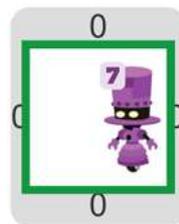
Tablero inicial



Tablero final



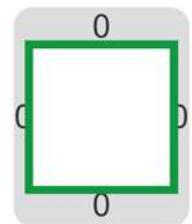
Tablero inicial



Tablero final



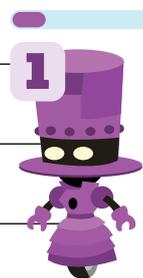
Tablero inicial



Tablero final

2. ¿Qué diferencia encontrarás entre los bloques `si [] / si no y` `si [] / si no, si [] / si no?`





NOMBRE Y APELLIDO:

CURSO:

FECHA:

AGREGAMOS PELOTAS, ¿O NO?



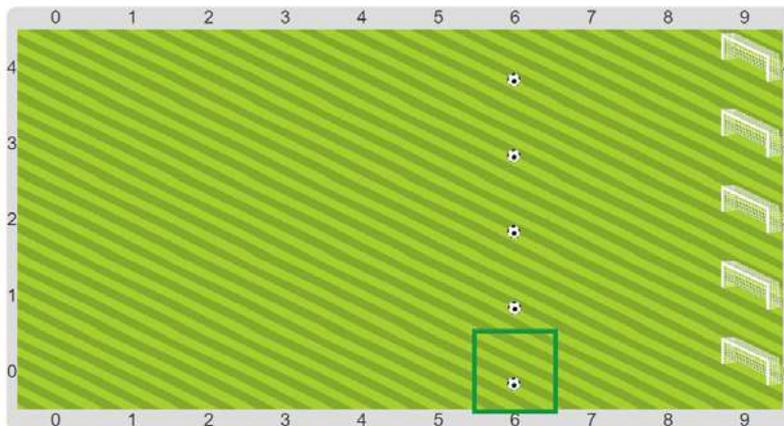
¡Hay que preparar el campo de juego para que el Beto pueda entrenar!

1. Abrió el proyecto "Agregamos pelotas, ¿o no?" y completá el programa para que al Beto le queden pelotas por patear a todos los arcos. Ya viene resuelto el procedimiento **Completar la columna de pelotas**. Vos tenés que ocuparte de completar el cuerpo principal del programa y el procedimiento **Agregar una pelota si hace falta**, que lo único que tiene que hacer es depositar una pelota en la celda bajo el cabezal solo si allí no hay una previamente. ¡Cuidado! Si agregás una pelota donde ya había una... ¡BOOM!

Definir **Agregar una pelota si hace falta**

COMPLETAR

Tenés que conseguir un tablero final como este:



Hay muchos tableros iniciales.
¡Tu programa tiene que
funcionar en todos!



2. Copiá acá abajo cómo completaste **Agregar una pelota si hace falta**:

BIBLIOTECA

En la biblioteca vas a encontrar disponibles el procedimiento **Agregar una pelota** y la función **hay una pelota**. ¡Usalos para resolver el desafío!

Agregar una pelota

hay una pelota

