

LightBot

LightBot es un videojuego educativo en donde se debe programar un robot que vive en una grilla cuadrículada y que tiene como misión encender luces, que se encuentran en diversos lugares del escenario, mientras sortea algunos obstáculos.

Vamos a programar una versión de LightBot en Gobstones. Primero explicaremos la dinámica del juego, y luego la representación a utilizar.

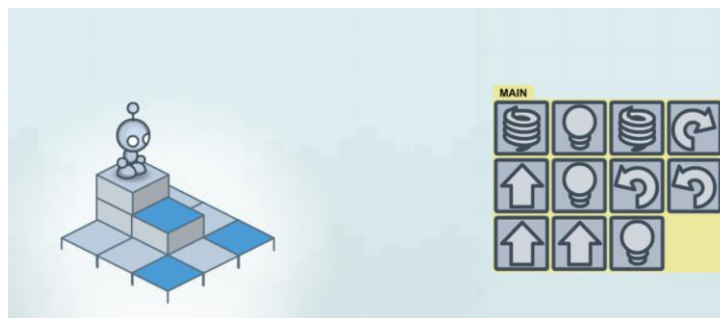
El **escenario** es una cuadrícula (rectangular, dividida en celdas), en donde cada celda tiene una altura (hay celdas con **altura** cero, otras con altura uno, otras con dos, etc.). Puede pensarse la altura como diferentes “escalones”.

Lightbot se encuentra en una única celda de esa cuadrícula, y puede moverse de una celda a otra de dos formas distintas: **caminando** de una a otra, lo cual solo puede hacer si la celda de inicio y la de destino tienen la misma altura; o **saltando** de una a otra, lo cual solo puede hacer si la celda de inicio y la de destino tienen una altura diferente, que varía en una unidad (es decir, se puede saltar de una celda de altura cero a una de altura uno, de uno a dos, etc. y también para bajar, de dos a uno, de uno a cero, etc. pero no se puede, por ejemplo, saltar de cero a dos, de dos a cero, ni de uno a uno).

Además LightBot solo sabe caminar “hacia adelante”, ya que **se encuentra siempre mirando hacia alguna dirección**. Si por ejemplo, se encuentra mirando al Norte, entonces al caminar, camina a la siguiente celda al Norte, y lo mismo para saltar. LightBot también sabe entonces **girar**, lo cual debe realizarse para cambiar la dirección a la que mira, pudiendo girar en sentido **horario** o **antihorario**.

En algunas celdas hay baldosas especiales, de color azul, que son **luces apagadas**. El objetivo del juego es que LightBot prenda esas luces (dejándolas de color amarillo), es decir, transformándolas en **luces prendidas**. Para esto LightBot puede **intercambiar el estado de una luz**, cambiando el estado de la misma de prendida a apagada y viceversa.

A continuación hay una pequeña imagen del juego original, para entender un poco más de qué va el mismo:



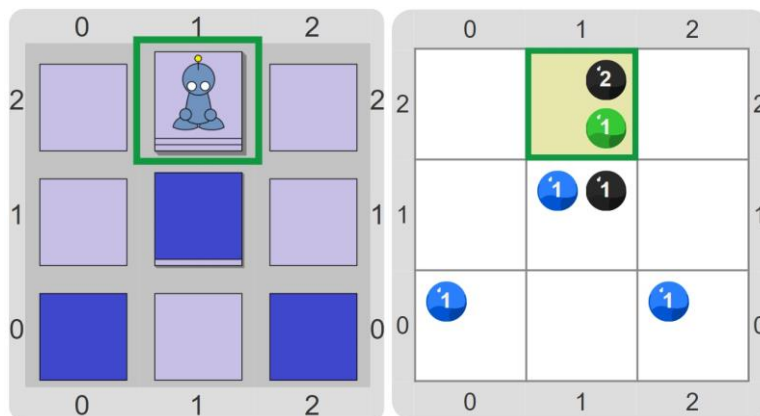
En el juego original se muestra a la izquierda el escenario, donde pueden apreciarse las distintas alturas del piso, a lightbot mirando hacia una dirección, y varias celdas azules, que deben ser prendidas. Del lado derecho están las acciones a realizar por LightBot para prender dichas luces (saltar, intercambiar el estado de una luz, volver a saltar, girar en sentido horario, avanzar, etc.)

En nuestra versión del juego no vamos a modelar la idea de que el usuario arma el programa, sino que solo tendremos el escenario, y los comandos a LightBot estarán dados por código Gobstones. Por supuesto, el escenario y las celdas van a estar dadas por el tablero en sí, y vamos a representar los elementos de la siguiente forma:

- La cantidad de **bolitas negras** en una celda indica la **altura** del piso. La altura más baja es cero, y no hay una cantidad máxima.
- Una **bolita azul** indica que hay una **luz apagada** en la celda, mientras que **dos bolitas azules** quiere decir que hay una **luz prendida**.
- **LightBot** está representado por **entre una y cuatro bolitas de color Verde**, dependiendo de para qué lado está mirando. Notar que “girar a lightbot consiste en cambiar la cantidad de bolitas verdes con la cual se representa). La dirección a la que mira está dada por:
 - Sí mira al Sur, entonces es una bolita verde.
 - Sí mira al Oeste son dos bolitas verdes.
 - Sí mira al Norte son tres bolitas verdes.

- Sí mira al Este son cuatro bolitas verdes.

A continuación se muestra como se vé el nivel de la imagen anterior en Gobstones, utilizando vestimentas, y también sin la vestimenta:



Puede apreciarse como LightBot está “mirando al Sur”, y como se encuentra en una plataforma de altura 2. También como la celda al Sur tiene altura 1, y como el resto tienen altura cero. Además pueden apreciarse tres luces apagadas, en las celdas (0,0), (1,1) y (2,0). No hay luces prendidas en este ejemplo.

El objetivo de este examen será programar una serie de operaciones para manipular a LightBot u otros elementos de este juego, u obtener información interesante, pero no vamos a programar el juego completo. Lea atentamente qué piden las actividades a continuación y realice sólo lo pedido.

Castillo A

- Ejercicio 1)** Se pide escriba el procedimiento **PrenderLuz** que, asumiendo que LightBot está en la celda actual y que en dicha celda hay una luz apagada, la prende. Se pide hacer también **ApagarLuz**, que hace lo opuesto, apagando la luz de la celda actual.
- Ejercicio 2)** Se pide escriba el procedimiento **MoverLightBotAlEste** que mueve a LightBot una celda hacia el Este de la actual. Se puede asumir que LightBot está en la celda actual mirando hacia el Este, que hay al menos una celda hacia el Este y que tanto la celda actual como la lindante al Este tienen la misma altura.
- Ejercicio 3)** Se pide escriba el procedimiento **Prender3ConsecutivasAlEste** que hace que LightBot prenda las luces de las siguientes 3 celdas consecutivas hacia el Este. Por supuesto puede asumirse que hay al menos tres celdas al Este, todas de la misma altura que la celda actual y que cada una de ellas tiene una luz apagada, además de que LightBot está en la celda actual mirando al Este.

Castillo B

Ejercicio 4) Se pide escriba el procedimiento **RotarEnSentidoHorario** que, asumiendo que LightBot está en la celda actual, gira a LightBot cambiando la dirección hacia la cual está mirando. Recordar que la representación de LightBot codifica la dirección a la que mira. Se pide hacer también **RotarEnSentidoAntihorario**.

Ejercicio 5) Se pide escriba el procedimiento **MoverLightBotAl_** que mueve a LightBot un paso en la dirección dada, asumiendo que LightBot se encuentra en la celda actual, mirando en dicha dirección. Para solucionar el problema puede hacer uso de la siguiente función primitiva:

codificaciónParaLightBotCuandoMiraAl_(direcciónDondeMira)

/*

PROPÓSITO: Describir la cantidad de bolitas verdes con la cual se representaría a LightBot si éste estuviera mirando en la dirección ****direcciónDondeMira****.

PARÁMETROS:

* **direcciónDondeMira:** Dirección - La dirección hacia la cual miraría LightBot y para la cual se desea saber la representación en bolitas verdes.

TIPO: Número

PRECONDICIONES: Ninguna.

*/

Ejercicio 6) Se pide escriba el procedimiento **Avanzar** que mueve a LightBot una celda hacia la dirección en la que mira. Puede asumir que LightBot se encuentra en la celda actual, y además puede hacer uso de la siguiente función primitiva:

direcciónALaQueMiraLightBot()

/*

PROPÓSITO: Describir la dirección a la cual mira LightBot.

TIPO: Dirección

PRECONDICIONES: LightBot se encuentra en la celda actual.

*/

Ejercicio 7) Se pide escriba la función **estáLightBotSobreLuzPrendidaAcá** que indica sí en la celda actual se encuentra LightBot y este está sobre una luz prendida.

Castillo C

- Ejercicio 8)** Se pide escriba el procedimiento **IrALightBot** que posiciona el cabezal sobre LightBot. Se puede asumir que LightBot está en algún lugar del tablero.
- Ejercicio 9)** Se pide escriba el procedimiento **ReiniciarNivel** que apaga todas las luces del tablero y posiciona a LightBot en la celda (0,0), sin importar donde estaba ubicado antes, pero mirando en la misma dirección que estaba mirando previo a reiniciar el nivel. Se asume que LightBot se encuentra en algún lugar del tablero.
- Ejercicio 10)** Se pide escriba la función **cantidadDeLucesPorPrenderEnAltura_** que dado un número describe la cantidad de luces que quedan aún por prender (las apagadas) y que están en la altura dada.