

Introducción a la Programación – UNQ – 1er semestre de 2009

PARCIAL 2 – SOKOBAN

Aclaraciones:

- Esta evaluación es a libro abierto. Se pueden usar todo lo visto en la práctica y en la teórica, aclarando la referencia.
- No se olvide de poner nombre, nro. de legajo, nro. de hoja y cantidad total de hojas en cada hoja.
- Le recomendamos leer el enunciado en su totalidad y organizar sus ideas antes de comenzar la resolución.
- Recuerde que la intención es medir cuánto comprende usted del tema. Por ello, no dude en escribir todo lo que sabe, en explicar lo que se propone antes de escribir código, en probar su código con ejemplos, etc.

El SOKOBAN es un juego donde un *Operario*¹ debe ordenar un cierto número de *Cajas* en un *Depósito*. El depósito consiste en una serie de pasillos delimitados por *Paredes* y con *Ubicaciones*, que son lugares indicados como que ahí debe ir una caja. El juego arranca con las cajas en cualquier lugar del depósito, y termina cuando el operario ha conseguido llevar todas las cajas a sus ubicaciones. La manera que tiene el operario de mover las cajas es pararse detrás de ellas, y empujarlas en alguna dirección, lo cuál sólo puede hacerse si el lugar detrás de la caja se encuentra vacío.

El objetivo de este parcial es modelar el SOKOBAN en C/GOBSTONES a través de un programa. Para ello, el tablero se utilizará para mostrar el depósito. Cada celda del tablero mostrará, o bien una pared, o bien una caja, o bien una ubicación, o bien la posición del operario. Obviamente, las paredes y las cajas no pueden ocupar las mismas celdas, las paredes y las ubicaciones tampoco, y el operario no puede estar ni sobre una pared ni sobre una caja; sí, en cambio, tanto el operario como una caja pueden compartir su lugar con una ubicación. El operario se mostrará con una bolita roja, las paredes con una bolita negra, las cajas con bolitas azules, y las ubicaciones con bolitas verdes.

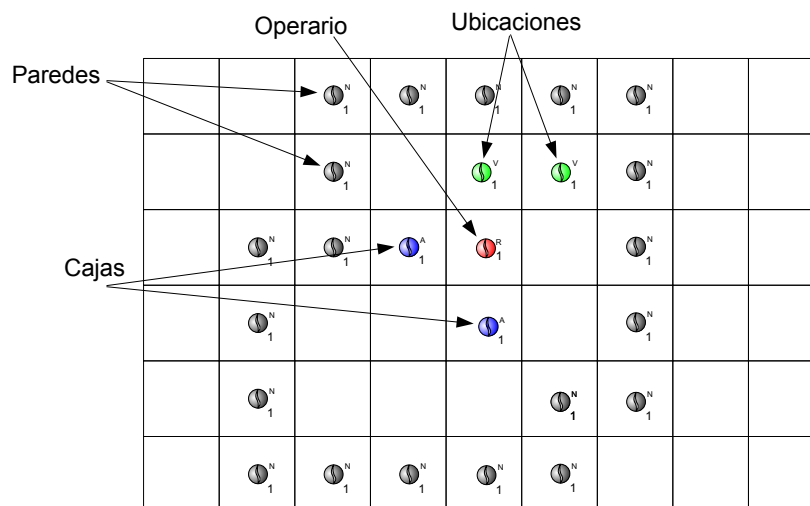


Figura 1: Tablero de ejemplo inicial del juego de SOKOBAN.

¹No, no se llama Mario, ni tiene bigote :P

Dado que modelar la mecánica del juego en el tablero es algo complejo, el mismo sólo se usará para mostrar el juego en determinados instantes. Para representar el juego se utilizarán registros y listas. Para ello, los siguientes ejercicios lo irán guiando en la modelización.

Ejercicio 1 Escribir, usando lógica de primer orden, un predicado *PuedeMover* que tome dos parámetros: una celda x y una dirección d , y sea verdadero si en x hay un operario y si el mismo se puede mover en la dirección d .

Ejercicio 2 Definir un tipo *Deposito* que permita representar un juego de SOKOBAN. El mismo debe contener representaciones para las paredes, las cajas, las ubicaciones, y el operario. Tenga en cuenta que puede usar representaciones auxiliares, como el tipo *Coord* usado en el ATOMIX. Escriba el invariante de representación correspondiente.

Ejercicio 3 Escribir una función *moverOperario* que dado un depósito y una dirección retorne un nuevo depósito donde el operario se movió en la dirección dada, si era posible. El operario no se puede mover hacia una pared, ni fuera del depósito, y si se mueve en dirección a una caja, sólo podrá moverse si la caja puede desplazarse a la celda que sigue en la dirección dada.

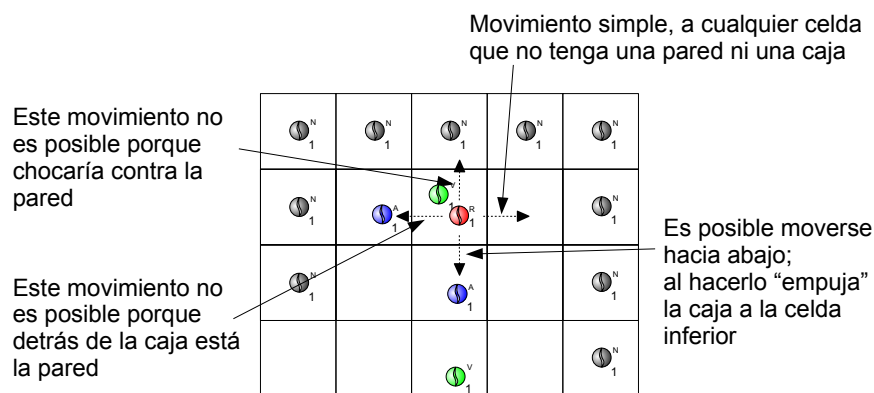


Figura 2: Movimientos posibles en el juego.

Ejercicio 4 Escribir una función *terminoTrabajo* que dado un depósito retorne un booleano que indica si el operario terminó su tarea o no

Ejercicio 5 Escribir un procedimiento *DibujarTablero* que dado un depósito lo muestre en el tablero, según la codificación explicada antes.