

Problema  
Grado en Robótica. Curso 2021-2022.  
Departamento de Electrónica y Computación

Nombre:.....

## 1. Problema

Una empresa dedicada a las máquinas expendedoras de comida tiene máquinas en varios edificios de la ciudad. Cada uno de estos edificios dispone de una única máquina expendedora. Además, esta empresa dispone de un robot capaz de comprobar el correcto funcionamiento de las máquinas. Para ello, cada día, el robot debe iniciar su ruta en el almacén, visitar cada edificio una única vez y, una vez ha visitado todos los edificios, volver de nuevo al almacén. La Figura 1 muestra los edificios con máquinas expendedoras (E1, E2 y E3) y el almacén de la empresa (AL). Sabiendo que el coste de moverse de un edificio a otro se denomina  $c_{i,j}$  donde  $i, j \in AL, E1, E2, E3$  y  $\forall i, j \ c_{i,j} = 1$  con  $i \neq j$ , y que se ha decidido modelar este problema como un problema de búsqueda se pide:

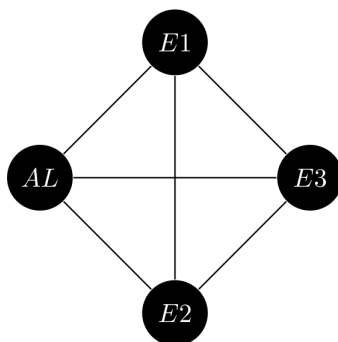


Figura 1: Plano de la ciudad.

1. ¿Cómo se puede representar cada estado? ¿cuál sería el estado inicial? ¿y final? (4 puntos)
2. ¿Cuáles son los operadores para la representación escogida? (4 puntos)
3. Suponiendo búsqueda en amplitud, ¿cuáles son los estados generados a partir del estado inicial? (2 puntos)

1. En este problema existen muchas representaciones para los estados. Una de ellas consiste en una tupla del tipo  $(PosR, E1, E2, E3)$  donde  $PosR$  es una variable que representa la posición del robot en el estado actual, con  $PosR \in \{ 'Al', 'E1', 'E2', 'E3' \}$ , y  $E1$ ,  $E2$  y  $E3$  se corresponden con variables booleanas que indican si el robot ha visitado o no ese edificio. Utilizando esta representación es fácil comprobar que el estado inicial sería  $(Al, 0, 0, 0)$  y el estado final  $(Al, 1, 1, 1)$ .
2. Los operadores o acciones son los que nos van a permitir transitar de un estado a otro. Por lo tanto, teniendo esto en mente, aunque nuevamente existen diferentes alternativas, podemos tener un único operador  $mover(x)$  donde la  $x$  se corresponde con la posición a la que el robot desea moverse. Las precondiciones y efectos se muestran en el siguiente procedimiento:

```
procedimiento mover(x):  
    if x == 'Al':  
        if E1 == 1 and E2 == 1 and E3 == 1:  
            # Solo podemos regresar al almacen si previamente  
            # se han visitado todos los edificios  
            PosR = 'Al'  
    elif x == 'E1' and E1 == 0:  
        PosR = 'E1'  
        E1 = 1  
    elif x == 'E2' and E2 == 0:  
        PosR = 'E2'  
        E2 = 1  
    elif x == 'E3' and E3 == 0:  
        PosR = 'E3'  
        E3 = 1
```

3. Como realmente nos piden expandir un único estado, el inicial, los sucesores resultantes de esta expansión van a ser los mismos tanto si utilizamos búsqueda en amplitud, profundidad,... La expansión de este nodo inicial  $(Al, 0, 0, 0)$  va a generar tres sucesores que se producen de tres posibles instanciaciones del operador  $mover(x)$ :  $mover('E1')$ ,  $mover('E2')$  y  $mover('E3')$ , que producen los sucesores  $(E1, 1, 0, 0)$ ,  $(E2, 0, 1, 0)$  y  $(E3, 0, 0, 1)$ , respectivamente. Habría una cuarta instanciación del operador  $mover(x)$ :  $mover('Al')$  pero puede comprobarse que esta instanciación no produciría ningún sucesor porque no se cumplen las precondiciones, es decir, para regresar al almacén es un requisito que el resto de edificios hayan sido visitados previamente, algo que evidentemente no se cumple en este estado inicial.