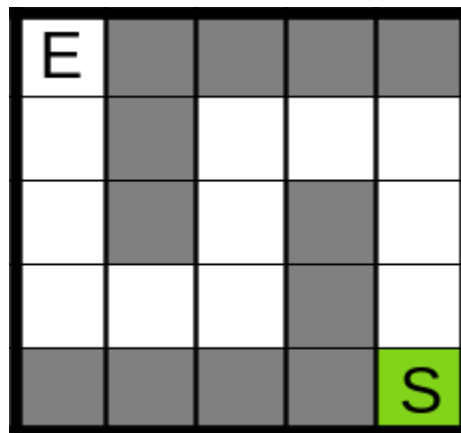


## Clásico Problema del Laberinto.

### Ejercicio 1:

Construir un laberinto como el mostrado en la figura siguiente:



El laberinto se puede representar como una matriz de filas x columnas (puede ser de entero o de char), cada casilla se representará con un espacio ' ' o con un numero cero si hay paso o con la letra 'X' o con el valor -1 si hay un muro, tal y como se muestra a continuación:

```
laberinto = {  
    {' ', 'X', 'X', 'X', 'X'},  
    {' ', 'X', ' ', ' ', ' '},  
    {' ', 'X', ' ', 'X', ' '},  
    {' ', ' ', ' ', 'X', ' '},  
    {'X', 'X', 'X', 'X', 'S'}  
}
```

```
laberinto = {  
    { 0, -1, -1, -1, -1},  
    { 0, -1, -1, -1, -1},  
    { 0, -1,  0, -1,  0},  
    { 0,  0,  0, -1,  0},  
    {-1, -1, -1, -1, -2}  
}
```

El laberinto se debe crear a partir de una tupla con las coordenadas de las casillas donde hay muro, como la siguiente:

```
muro = ((0,1), (0,2), (0,3), (0,4), (1,1), (2,1),  
(2,3), (3,3), (4,0), (4,1), (4,2), (4,3))
```

## Ejercicio 2:

Construir un programa para recorrer el laberinto desde la entrada a la salida. La entrada siempre está en la esquina superior izquierda y la salida en la esquina inferior derecha.

El programa debe devolver una matriz de string con la secuencia de movimientos para ir de la entrada a la salida del laberinto, tal y como se muestra a continuación:

```
{'Abajo', 'Abajo', 'Abajo', 'Abajo', 'Derecha',  
'Derecha', 'Arriba', 'Arriba', 'Derecha', 'Derecha',  
'Abajo', 'Abajo', 'Abajo'}
```