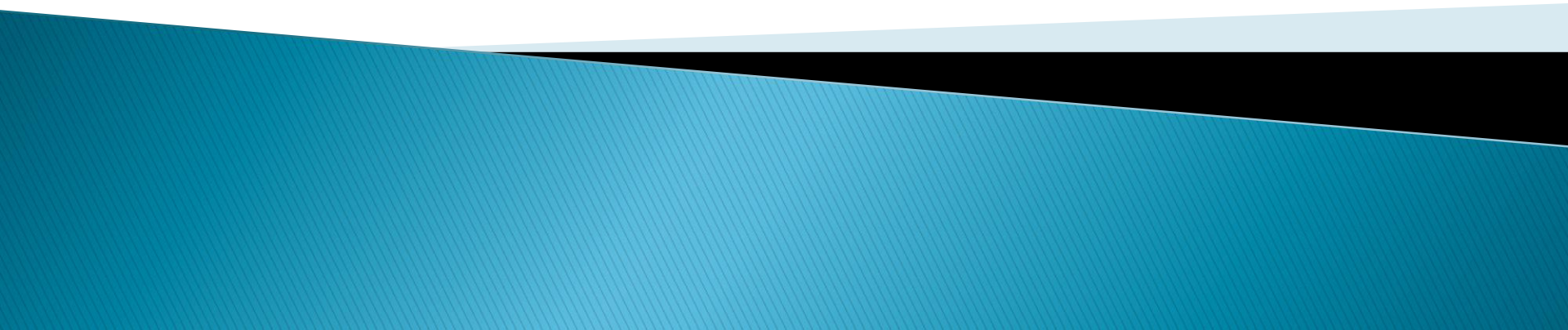


# Práctica 4: Algoritmos de vuelta atrás (Backtracking)

Laberinto

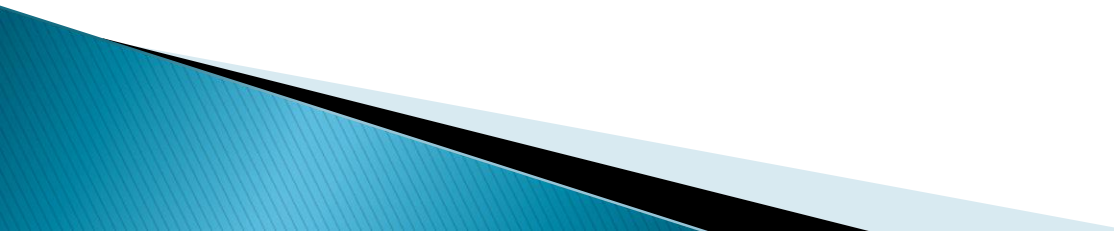


# 1. Enunciado del problema

- ▶ El problema consiste en encontrar la salida de un laberinto.
- ▶ Se representa con una matriz bidimensional de tamaño  $n \times n$ .
- ▶ Cada posición almacena un valor 0 si la casilla es transitable y cualquier otro valor si la casilla no es transitable.
- ▶ Los movimientos permitidos son a casillas adyacentes de la misma fila o la misma columna.
- ▶ Podemos suponer que las casillas de entrada y salida del laberinto son  $(0,0)$  y  $(n-1, n-1)$  respectivamente.
- ▶ Deberemos implementar un algoritmo backtracking para resolver el problema.
- ▶ Modificar el algoritmo para que encuentre el camino más corto.
- ▶ Realizaremos también un estudio empírico de la eficiencia del algoritmo.

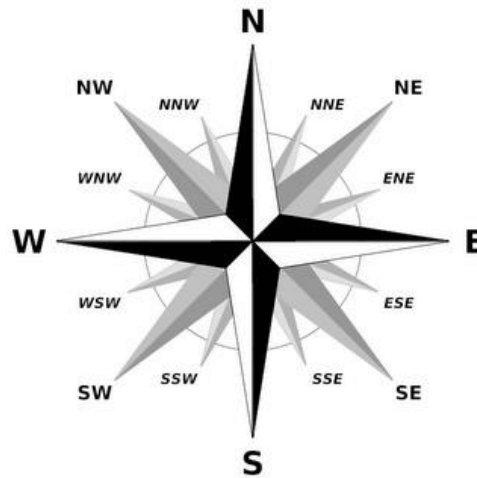


## 2. Introducción

- ▶ El backtracking es una estrategia usada para encontrar soluciones a problemas que tienen una solución completa,
  - ▶ Esta estrategia se asemeja a un recorrido en profundidad dentro de un grafo dirigido. El grafo en cuestión suele ser un árbol, pero esta estructura existe sólo implícitamente.
  - ▶ De forma general, el método del backtracking, genera todas las secuencias de forma sistemática y organizada, de manera que prueba todas las posibles combinaciones construyendo soluciones parciales hasta llegar a la correcta.
  - ▶ La forma de actuar consiste en elegir una alternativa del conjunto de opciones de cada etapa del proceso de resolución, y si esta elección no funciona (no nos lleva a ninguna solución), la búsqueda vuelve al punto donde realizó dicha elección, e intenta con otro valor. Cuando se han agotado todos los posibles valores en ese punto, la búsqueda vuelve a la anterior fase en la que se hizo otra elección entre valores, Si no hay más puntos de elección, la búsqueda finaliza.
- 

# 3. Backtracking en el problema del laberinto

- ▶ Primero crea un laberinto con los valores correspondientes, de 0 en las casillas que son transitables y uno las que no.
- ▶ Primero se toma la primera decisión que será moverse desde la casilla  $[0, 0]$  hacia el este o si no puede va hacia el sur.
- ▶ El movimiento lo va alternando el movimiento en las agujas del reloj.
- ▶ De esta manera va avanzando el camino a la salida, si en algún momento la rama en la que nos encontramos no llega a una solución vuelve atrás por la recursividad y sigue explorando el espacio de soluciones hasta dar con la solución adecuada.
- ▶ Para la mejor solución va guardando el número de recorridos y cogiendo el menor.



# 4. Ejemplos de soluciones de laberintos.

Tama: 10  
Entrada: (0 , 0)  
Salida: (9 , 9)

```
0 1 1 0 0 0 1 1 0 0
0 0 0 0 0 0 1 0 0 1
0 1 0 1 1 1 1 0 1 1
0 0 0 0 1 1 0 0 0 0
1 1 1 1 1 1 1 0 1 1
1 1 0 1 1 1 0 0 1 0
0 1 0 0 1 1 1 1 0 0
0 1 1 1 0 0 0 1 0 0
0 1 1 0 0 0 1 1 1 0
1 1 1 1 1 1 1 1 0 0
```

Solucion:

No hay solucion

MEJOR Solucion:

No hay solucion

Tama: 6  
Entrada: (0 , 0)  
Salida: (5 , 5)

```
0 0 0 0 0 0
0 1 1 1 1 0
0 0 0 0 0 0
0 1 1 1 1 1
0 0 0 1 1 1
0 0 0 0 0 0
```

Solucion:

E --> E --> E --> E --> E --> S --> S --> 0 --> 0 --> 0 --> 0 --> 0 --> S --> S --> E --> E --> S --> E --> E --> E -->

MEJOR Solucion:

S --> S --> S --> S --> E --> E --> S --> E --> E --> E -->

Tama: 10  
Entrada: (0 , 0)  
Salida: (9 , 9)

```
0 1 1 1 0 1 1 1 1 0
0 0 1 1 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 1
1 0 1 0 1 1 0 0 1 0
0 0 0 0 1 1 1 1 1 1
1 0 1 1 1 0 0 1 1 0
0 0 0 1 0 0 0 0 0 1
0 0 0 0 0 1 1 1 0 0
0 1 1 1 1 0 0 1 1 0
```

Solucion:

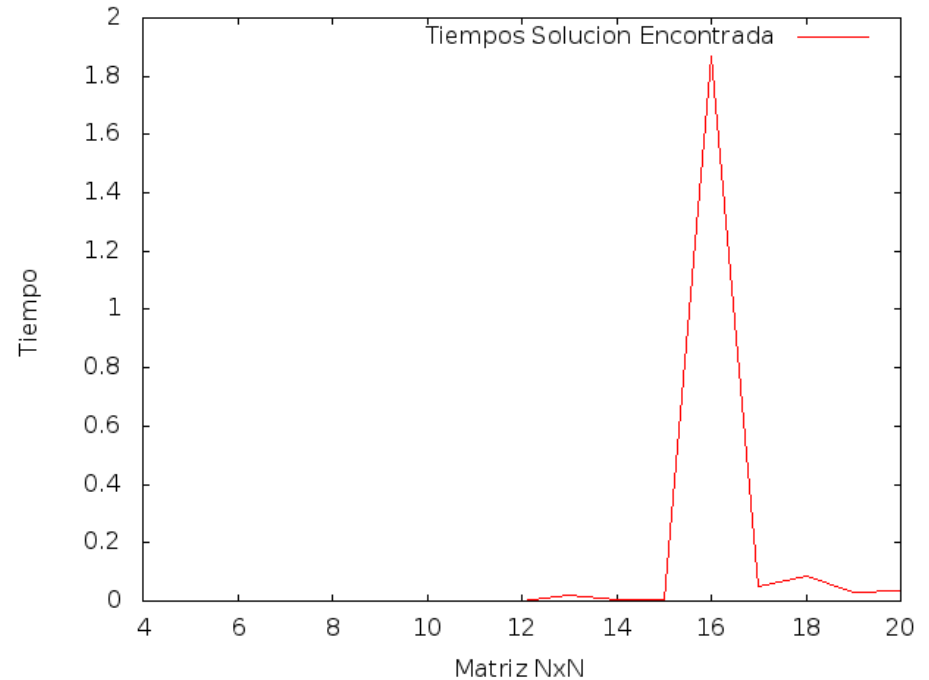
S --> E --> S --> E --> E --> S --> S --> S --> 0 --> 0 --> S --> S --> E --> S --> E --> E --> N --> E --> E --> E --> S --> E --> S -->

MEJOR Solucion:

S --> E --> S --> S --> S --> S --> S --> S --> E --> S --> E --> E --> N --> E --> E --> E --> E --> S --> E --> S -->

# 5. Cálculo de la eficiencia empírica.

Elementos matriz	Tiempo(s)	Elementos matriz	Tiempo(s)
4	3,43E-03	13	0.0199105
5	2,96E-02	14	0.00431616
6	9,83E-03	15	0.00699563
7	3,19E-03	16	187.063
8	5,40E-01	17	0.0521821
9	4,19E-01	18	0.0874728
10	0.00114296	19	0.031764
11	0.00164059	20	0.0330296
12	0.000128574		



Con elementos de la matriz nos referimos a el numero de filas y columnas de la matriz utilizada.

# 6. Eficiencia empírica. Algoritmo mejorado.

Elementos matriz	Tiempo(s)	Elementos matriz	Tiempo(s)
4	1,80E+00	13	0.00139031
5	9,50E+00	14	0.0825609
6	0.000140302	15	0.00549129
7	0.000645815	16	0.164093
8	0.000541086	17	0.00131956
9	0.00197997	18	0.0122801
10	0.00351746	19	0.379662
11	0.000585826	20	0.264537
12	0.000698876		

