

# Recursión

Divide y vencerás

Verónica E. Arriola-Rios

Facultad de Ciencias, UNAM

5 de septiembre de 2023



# Divide y vencerás

- 1 Divide y vencerás
- 2 Retroceso (*Backtracking*)
- 3 Implementación
- 4 Bibliografía

# Divide y vencerás

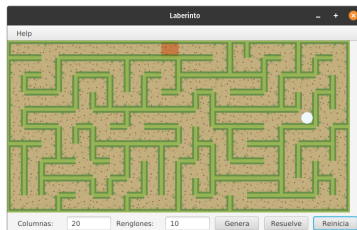
- La estrategia *divide y vencerás* propone la solución de problemas complejos mediante su descomposición en subproblemas más simples.
- En el caso de la programación recursiva estos subproblemas son:
  - Caso base.* El caso más sencillo, para el cual se conoce la solución.
  - Caso recursivo.* En el cual se plantea la solución al problema en términos de la solución a un subproblema o ejemplar más sencillo del mismo problema, más algunos pasos sencillos que completan la solución para el nuevo ejemplar.

## Retroceso (*Backtracking*)

- 1 Divide y vencerás
- 2 Retroceso (*Backtracking*)
- 3 Implementación
- 4 Bibliografía

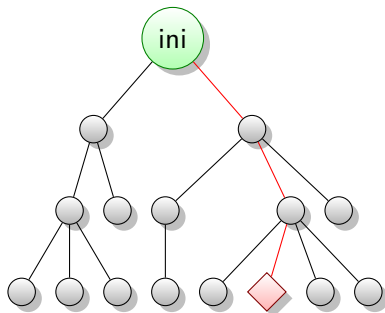
# Retroceso

- **Retroceso**, o *backtracking* en inglés, es un algoritmo de búsqueda para la resolución de problemas.
- El algoritmo se ejecuta sobre un espacio donde se muestran las posibles acciones que se pueden tomar para llegar a un objetivo dado.
- El resultado devuelto debe ser una secuencia de acciones que indiquen cómo llegar del punto inicial a la solución del problema, en caso de haber solución, nada si no la hay.

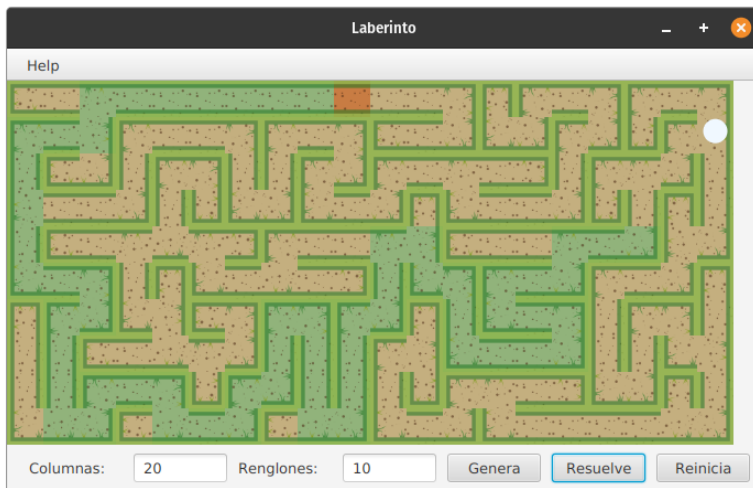


# Concepto

- Conceptualmente, retroceso busca una hoja solución en un árbol de planes posibles.
- Cada ruta que intenta es el camino del punto inicial (raíz) hacia alguna hoja.
- Sólomente cuando el algoritmo descende hasta una hoja solución se dice que el problema está resuelto.



# Ejecución y resultado



# Implementación

- 1 Divide y vencerás
- 2 Retroceso (*Backtracking*)
- 3 Implementación**
- 4 Bibliografía

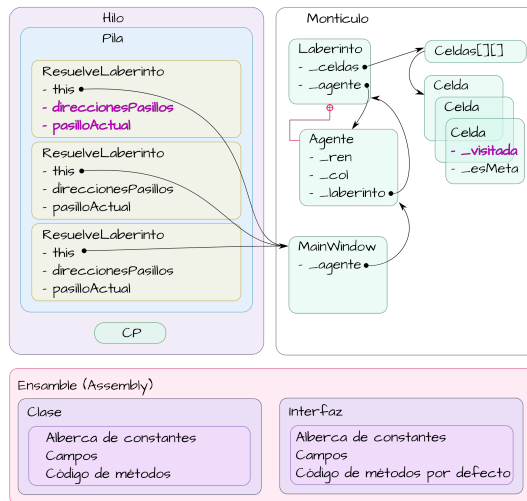


# Uso de la memoria

Para implementar este algoritmo de forma recursiva se utilizarán dos formas de memoria:

- *Marcado de celdas*, el agente tendrá la posibilidad de marcar o desmarcar la celda en la que se encuentre parado.
- Los *registros* de llamadas a **métodos** en la **pila de ejecución** de C#.
  - Recuérdese que aquí se guardan las variables locales y **resultados intermedios**.
  - Los valores de estas variables permanecen mientras se llama a otros métodos y se regresa de ellos.


# En la máquina virtual



## Bibliografía

- 1 Divide y vencerás
- 2 Retroceso (*Backtracking*)
- 3 Implementación
- 4 Bibliografía

## Bibliografía I

-  Buck, Jamis (feb. de 2011). *Maze Generation: Algorithm Recap*. URL: <http://weblog.jamisbuck.org/2011/2/7/maze-generation-algorithm-recap>.

# Licencia

Creative Commons  
Atribución-No Comercial-Compartir Igual

