Recursividad

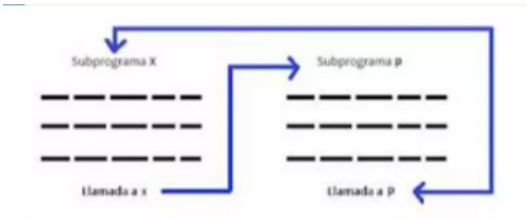
DEFINICIONES

Es una función que se llama a sí misma para resolver un problema. Se usa para escribir código más elegante y ordenado, dividiendo un problema grande en subproblemas más pequeños. Es muy útil en casos como cálculos matemáticos o estructuras repetitivas. Sin embargo, puede consumir mucha memoria y a veces es más lenta que una solución con bucles (iterativa).

DEFINICION

RECURSIVIDAD INDIRECTA

- La función A llama a la función B, y B llama de nuevo a A.



Recursion vs Iteracion

Recursion

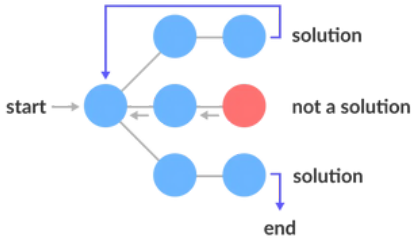
- Mas clara para problemas repetitivos
- Usa pila de llamadas
- Riesgo de stack overflow

Iteracion

- Más eficiente en memoria
- Usa bucles (for, while)
- Control directo de memoria

BACKTRACKING

Es una técnica donde se prueba todas las soluciones posibles y se retrocede cuando una solución no funciona para encontrar una mas adecuada.



COMPONENTES

Caso base

- Condición que detiene la recursión.
- Sin él, la función se llamaría infinitamente.

Llamada recursiva

- La función se llama a sí misma con un problema más pequeño.

PILA DE LLAMADAS

- Cada llamada recursiva se apila en memoria. La pila guarda el valor actual de las variables la posición donde el programa debe continuar después
- Se resuelve en orden LIFO (Last In, First Out).
- Si la pila se llena → Stack Overflow. (demasiadas llamadas recursivas)