

### Práctica #4 - Backtracking

1. Describa el concepto de recursividad ¿Qué significan los ambientes recursivos?
2. Explique brevemente los estados (variables y términos de alcance) creados en cada llamada recursiva en un algoritmo de backtracking.
3. ¿Cuáles son los principales esquemas de solución utilizados en backtracking? ¿Cuál es la diferencia con el esquema de fuerza bruta?
4. Dado un arreglo de números enteros y un valor entero  $k > 0$ , construya un algoritmo que imprima todas las formas posibles de obtener el valor  $k$  mediante la suma de los elementos del arreglo. Por ejemplo, para el arreglo  $arr = \{1, 2, 3, 4, 5\}$  y  $k = 5$ ; las formas posibles son  $1 + 4$ ,  $2 + 3$  y  $5$ .
5. Dada una matriz de dimensiones  $n \times n$  compuesta por números enteros, elabore un algoritmo que determine la sub-matriz de suma máxima.
6. Un pastor se encuentra en la orilla de un río con una oveja, un lobo y una col, para pasar a la otra orilla dispone de una barca muy pequeña en la que solo puede cruzar acompañado como máximo por uno de los tres. Realice un algoritmo que encuentre el modo para que el pastor pueda trasladar a la oveja, al lobo y la col a la otra orilla de manera exitosa. Se debe tener en cuenta como condición que el pastor no debe dejar a solas al lobo con la oveja, ni a la oveja con la col.
7. Escriba un algoritmo que permita conocer todas las formas en las cuales un juego de dominó está trancado. Un juego de dominó se considera "trancado" (solo ocurre en parejas), si las dos piezas que permiten a los jugadores participar en el juego son iguales, aún quedan fichas en posesión de los cuatro jugadores pero ninguno puede jugar.
8. Realice un algoritmo que permita generar todas las permutaciones posibles de un arreglo de caracteres de tamaño  $n$ . Luego, elabore otro algoritmo que permita generar las permutaciones del mismo arreglo, asegurando que la suma sea un número impar.
9. Dados dos números enteros  $n$  y  $r$ , elabore un algoritmo que permita imprimir todas las expresiones aritméticas compuestas por los operadores  $(+ - * /)$  y los primeros  $n$  números naturales en orden, donde el resultado sea igual a  $r$  (orden de prioridad es de izquierda a derecha). Por ejemplo, para  $n=4$  y  $r=0$ , algunas expresiones posibles son  $-1 -2 +3 * 4 = 0$ ,  $+1 +2 -3 *4 = 0$ .
10. Elabore un algoritmo que indique todas las formas en las cuales la pieza de caballo colocada en la esquina superior derecha de un tablero de ajedrez puede recorrerlo completo, sin pasar dos veces por el mismo lugar. Tenga en cuenta que el tablero de ajedrez es de tamaño  $8 \times 8$ , y el caballo se mueve en forma de "L". Luego, elabore un algoritmo que indique si el caballo puede recorrer el tablero sin pasar dos veces por el mismo lugar.
11. Realice un algoritmo que permita obtener todos los números de  $m$  cifras (menor o igual a nueve) distintas de cero y distintas entre sí, de tal manera que el número formado por las primeras  $n$  cifras (menor o igual a  $m$ ) sea múltiplo de  $n$ . Por ejemplo, para  $m = 4$  algunos resultados posibles son 1236, ya que 1 es múltiplo de 1, 12 de 2, 123 de 3 y 1236 de 4 y 9872, pues 9 es múltiplo de 1, 98 de 2, 987 de 3 y 9872 de 4.

12. Realice un algoritmo que permita colorear un mapamundi (en caso de ser posible). La manera de colorear el mapa es la siguiente, para que un color sea válido para un determinado país debe cumplirse que sea distinto del que tienen asignado todos sus países limítrofes. Se debe utilizar la menor cantidad de colores posibles. Asuma que existen  $n$  países distintos, y que se cuenta con una matriz de dimensiones  $n \times n$  de tipo bool, que indica si un país es limítrofe de otros. Por último, se debe considerar que los colores se representan mediante enteros.
13. En un tablero de ajedrez de dimensiones  $n \times n$  consideramos el siguiente juego. En el tablero hay colocados peones blancos y negros. Partiendo de una posición inicial y realizando movimientos válidos deseamos saltar todos los peones blancos de acuerdo con las siguientes reglas:
- Solo se permiten movimientos en cruz.
  - Tipos de movimientos posibles:
    - i. Movimiento a una casilla vacía. Coste de longitud 1.
    - ii. Salto de un peón blanco. Coste de longitud 2.
    - iii. No se pueden saltar peones negros.

Escriba un algoritmo que determine si dada una posición inicial y un número máximo de movimientos, es posible saltar a todos los peones blancos. Realice el árbol de estados creados, la llamada inicial (en el programa principal) y la llamada recursiva en el algoritmo completo.

14. Escriba un algoritmo que permita colocar  $k$  reinas en un tablero de ajedrez de dimensiones  $k \times k$ , sin que se ataquen entre ellas. En el ajedrez, una reina puede "atacar" a otra pieza desplazándose cualquier distancia en casillas: sobre la fila que ocupa, la columna que ocupa o bien sobre las diagonales que ocupa.
15. En una ferretería se venden segmentos de cable, los clientes suelen pedir varios metros de cable, por lo que siempre quedan pedazos cortos de cable sin vender. Por ello se ha decidido soldar todos los pedazos para vender cables de 100 metros, a un precio razonablemente menor. El ferretero entiende que este problema es complejo, pues tiene que tomar un conjunto que pedazos de cable que sumen exactamente 100 metros. Por esa razón el ferretero recurre a los estudiantes de AYED para que le resuelvan su problema, que en resumen consiste en: dada las longitudes de cada uno de los  $n$  pedazos de cable existentes, indicar si se puede generar un cable de 100 metros, y en dado caso, saber qué segmentos de cable se soldaron para generar uno de 100 metros con la menor cantidad de soldaduras posibles, para mantener los bajos costos.