Actividad 3 Diseño de Distribución de Equipos Electrónicos en Oficinas En una oficina de 4x4, se necesita organizar la disposición de computadoras y impresoras de manera que optimice el uso del espacio y facilite el acceso. Las restricciones de diseño son las siguientes: No puede haber dos computadoras en la misma fila o columna. No puede haber dos impresoras en la misma fila o columna. Debes encontrar todas las combinaciones posibles para colocar 4 computadoras y 4 impresoras en el tablero, respetando las restricciones anteriores. Objetivo: Implementar un programa en Java que utilice la técnica de backtracking para encontrar todas las configuraciones posibles de colocación de computadoras e impresoras en el tablero de 4x4. Tu programa debe imprimir cada configuración válida.

public class DistribucionEquiposOficina {

private static final int TAMANO\_OFICINA = 4;

private static char[][] oficina = new char[TAMANO\_OFICINA][TAMANO\_OFICINA];

public static void main(String[] args) {

System.out.println("Combinaciones válidas de computadoras (C) e impresoras (I) en una oficina de 4x4:");

encontrarCombinacionesValidas(0);

}

private static void encontrarCombinacionesValidas(int fila) {

if (fila == TAMANO\_OFICINA) {

imprimirOficina();

return;

}

for (int columna = 0; columna < TAMANO\_OFICINA; columna++) {

if (esPosicionValida(fila, columna, 'C')) {

oficina[fila][columna] = 'C'; // Colocar computadora

if (colocarImpresoras(fila)) {

encontrarCombinacionesValidas(fila + 1);

}

oficina[fila][columna] = '\0'; // Retroceder (Backtracking)

}

}

}

private static boolean colocarImpresoras(int fila) {

if (fila == TAMANO\_OFICINA) {

return true;

}

for (int columna = 0; columna < TAMANO\_OFICINA; columna++) {

if (esPosicionValida(fila, columna, 'I')) {

oficina[fila][columna] = 'I'; // Colocar impresora

if (colocarImpresoras(fila + 1)) {

return true;

}

oficina[fila][columna] = '\0'; // Retroceder (Backtracking)

}

}

return false;

}

private static boolean esPosicionValida(int fila, int columna, char elemento) {

// Verificar si la fila o columna ya tienen un elemento del mismo tipo

for (int i = 0; i < TAMANO\_OFICINA; i++) {

if (oficina[fila][i] == elemento || oficina[i][columna] == elemento) {

return false;

}

}

return true;

}

private static void imprimirOficina() {

for (char[] fila : oficina) {

for (char celda : fila) {

System.out.print((celda == '\0' ? "-" : celda) + " ");

}

System.out.println();

}

System.out.println();

}

}