



TAREA 1

UNIDAD II

MARIA GUADALUPE BARRON ZAZUETA

TOPICOS DE IA

1. Problema de Programación de Trabajos (JSSP)

Definición:

El **Job Shop Scheduling Problem (JSSP)** es un problema de optimización en el que se deben programar un conjunto de trabajos en un conjunto de máquinas, minimizando el tiempo total de finalización (makespan) o maximizando la eficiencia del sistema.

Representación:

- Se utiliza una matriz donde las filas representan los trabajos y las columnas representan las máquinas.
- Cada celda indica el tiempo que un trabajo necesita en una máquina específica.
- Se pueden usar diagramas de Gantt para visualizar la secuencia de trabajos en cada máquina.

Reglas y restricciones:

- Cada trabajo tiene una secuencia predefinida de máquinas por las que debe pasar.
- Una máquina solo puede procesar un trabajo a la vez.
- Un trabajo no puede iniciarse en una máquina hasta que haya terminado en la anterior de su secuencia.

Aplicaciones:

- Planificación de producción en fábricas.
- Programación de tareas en supercomputadoras.
- Gestión de proyectos con múltiples dependencias.

Tipos de problemas similares:

- Flow Shop Scheduling (cuando todos los trabajos siguen la misma secuencia de máquinas).
- Open Shop Scheduling (cuando los trabajos pueden procesarse en cualquier orden).

2. Problema de las N Reinas

Definición:

El **problema de las N reinas** consiste en colocar **N reinas en un tablero de ajedrez de $N \times N$** sin que se ataquen entre sí.

Representación:

- Se puede representar como una matriz de $N \times N$, donde cada celda representa una casilla del tablero.
- Otra forma de representación más eficiente es con un vector de tamaño N , donde el índice representa la columna y el valor la fila donde está colocada la reina.

Reglas:

- No puede haber dos reinas en la misma fila.
- No puede haber dos reinas en la misma columna.
- No puede haber dos reinas en la misma diagonal.

Aplicaciones:

- Problemas de optimización combinatoria.
- Inteligencia artificial (búsqueda y heurísticas).
- Algoritmos de retroceso y programación genética.

Técnicas de solución:

- **Backtracking (búsqueda exhaustiva)**
- **Algoritmos genéticos**
- **Métodos heurísticos (algoritmo de Las Vegas, Hill Climbing)**

3. Árbol de Expansión Mínima (MST - Mínimo Spanning Tree)

Definición:

Un **árbol de expansión mínima** es un subconjunto de un gráfico ponderado que:

- **Conecta todos los nodos sin formar ciclos .**
- **Minimiza la suma de los pesos de las aristas .**

Representación:

- Se usa un **grafo ponderado** donde los vértices representan nodos y los aristas tienen un peso asociado.
- Se pueden representar con listas de adyacencia o matrices de adyacencia.

Algoritmos para encontrar el MST:

1. **Algoritmo de Prim** : Expande un conjunto de nodos conectados con el menor costo posible.
2. **Algoritmo de Kruskal** : Ordena todas las aristas por peso y construye el MST agregando las de menor costo sin formar ciclos.
3. **Algoritmo de Borůvka** : Forma componentes conectados y los une iterativamente.

Aplicaciones:

- Diseño de redes (eléctricas, telecomunicaciones).
- Algoritmos de compresión de datos (como en JPEG y clustering).
- Planificación de rutas óptimas.

4. Problema del Agente Viajero (TSP - Problema del viajante)

Definición:

El **problema del agente viajero** consiste en encontrar la ruta más corta para que un vendedor visite **N ciudades una sola vez y regrese a la ciudad de origen**.

Representación:

- Se modela como un **gráfico completo ponderado** donde:
 - Los **vértices** representan ciudades.
 - Los **aristas** representan las distancias o costos de viaje entre ciudades.

Reglas:

- Se debe visitar cada ciudad exactamente una vez.
- Se debe minimizar la distancia total recorrida.
- Se debe regresar a la ciudad de inicio.

Métodos de solución:

- **Algoritmo de fuerza bruta ($O(n!)$)** : Evalúa todas las posibles rutas (impráctico para grandes valores de N).
- **Algoritmo de programación dinámica (Held-Karp, $O(n^2 \cdot 2^n)$)** : Optimiza el recorrido con subproblemas.
- **Algoritmos heurísticos y metaheurísticos** :
 - Algoritmo de Vecino Más Cercano.
 - Algoritmos Genéticos.
 - Recocido Simulado.
 - Búsqueda Tabú.

Aplicaciones:

- Optimización de rutas de transporte y logística.
- Diseño de circuitos en microchips.
- Planificación de entregas en servicios de paquetería.
- Secuenciación del ADN en bioinformática.