5.4. Reducciones vía gadgets

Definición (Hamiltonian Cycle)**. Input:** Un grafo dirigido G.
Output: ¿Hay un ciclo que visite cada vértice exactamente una vez?

$$\text{HAMCYCLE} = \{\langle G\rangle : G \text{ tiene un ciclo hamiltoniano}\}.$$

Teorema. HAMCYCLE es NP-completo.

Definición (3-Coloring)**. Input:** Un grafo no dirigido G.
Output: ¿Se pueden colorear los vértices con a lo más tres colores, sin vecinos del mismo color?

$$\text{3COLOR} = \{\langle G\rangle : G \text{ admite una 3-coloración}\}.$$

Teorema. 3COLOR es NP-completo.

Definición (3D-Matching)**. Input:** Conjuntos disjuntos X, Y, Z de tamaño n, y $T \subseteq X \times Y \times Z$.
Output: ¿Hay n ternas de T que cubran cada elemento exactamente una vez? Un conjunto así es un *3d-matching perfecto*.

$$\text{3DM} = \{\langle X, Y, Z, T\rangle : T \text{ admite un 3d-matching perfecto}\}.$$

Teorema. 3DM es NP-completo.

Definición (Subset Sum)**. Input:** Números naturales $w_1, \ldots, w_n, W$ en binario.
Output: ¿Hay un subconjunto de los w_i que sume W?

$$\text{SUBSET-SUM} = \{\langle w_1, \ldots, w_n, W\rangle : \exists I \subseteq \{1, \ldots, n\},\ \sum_{i\in I} w_i = W\}.$$

Teorema. SUBSET-SUM es NP-completo.

Definición (TSP)**. Input:** Distancias d entre n ciudades y una cota D.
Output: ¿Hay un tour de largo $\leq D$ que visite cada ciudad una vez y regrese al inicio?

Definición (Set Cover)**. Input:** Una familia $\mathcal{S}$ de subconjuntos de U y una cota k.
Output: ¿Hay a lo más k conjuntos de $\mathcal{S}$ cuya unión sea U?

Definición (Knapsack)**. Input:** Pesos w_i, valores v_i $(i = 1, \ldots, n)$ y cotas W, V, todos naturales en binario.
Output: ¿Hay $I \subseteq \{1, \ldots, n\}$ con $\sum_{i\in I} w_i \leq W$ y $\sum_{i\in I} v_i \geq V$?

Ejercicios

1. Formule cada problema como un lenguaje formal y demuestre que es NP-completo.

   a) Dado un grafo no dirigido G, decidir si sus vértices pueden colorearse con a lo más cuatro colores, sin vecinos del mismo color.

   b) Dados cuatro conjuntos disjuntos W, X, Y, Z, cada uno de tamaño n, y $T \subseteq W \times X \times Y \times Z$, decidir si hay n cuádruplas de T que cubran cada elemento exactamente una vez.

   c) Dado un grafo dirigido G y dos vértices s, t, decidir si hay un camino de s a t que visite cada vértice de G exactamente una vez.

   d) Dados enteros positivos $w_1, \ldots, w_n$ y números naturales W, k, todos en binario, decidir si hay exactamente k índices cuyos pesos sumen W.

2. Usted trabaja en una empresa de desarrollo de software y recibe los siguientes problemas de sus clientes. ¿Cuál sería su recomendación profesional? Justifique su respuesta con un argumento de complejidad computacional.

   a) Una consultora de ciencia de los datos quiere resolver un problema de *clustering* en un conjunto de n objetos. Para cada par de objetos i, j la consultora posee una medida de disimilitud $d_{ij} \geq 0$ que satisface $d_{ii} = 0$ y $d_{ij} = d_{ji}$. La consultora desea un algoritmo que decida si se pueden agrupar los objetos en a lo más tres grupos de manera que la distancia entre cualquier par de objetos del mismo grupo sea a lo más B.

   b) Un laboratorio debe ejecutar n tareas en una sola máquina. Cada tarea i tiene un instante a partir del cual está disponible r_i, una duración $p_i > 0$ y una fecha límite d_i; (todos son enteros no negativos escritos en binario). Cada tarea debe ejecutarse de corrido durante p_i unidades de tiempo, comenzar a partir de r_i y terminar a más tardar en d_i.

   El laboratorio quiere un algoritmo que decida si se pueden completar todas las tareas respetando esas condiciones.