## Parte D: Algoritmos Golosos

1. El problema continuo de la mochila. Tenemos $n$ sustancias, cada una de ellas con peso $w_i$ y valor $v_i$ por unidad. De la sustancia $i$ tenemos un total $a_i$ (no necesariamente entero) de unidades. A diferencia del problema discreto de la mochila, ahora podemos transportar fracciones de unidades. Para maximizar el valor total de la carga sujeto a que el peso sea menor que un valor $P$ dado, se propone el siguiente algoritmo:

   *a*) Calcular la "densidad" de valor" de cada sustancia, $d_i = v_i/w_i$.

   *b*) Ordenar las sustancias en función de $d_i$.

   *c*) De la sustancia con mayor $d_i$, cargar la mayor cantidad posible (o sea, hasta agotar $a_i$ o bien $P$).

   *d*) Repetir lo mismo con la segunda sustancia y así.

   Demuestre que este algoritmo es óptimo.

2. Dado un grafo $G = (V, W)$ conexo con pesos en las aristas, un **arbol generador** es un árbol $T = (V', W')$ tal que $V' \subset V$ y $W' \subset W$. Su peso es la suma de sus aristas. Un **árbol generador mínimo** es un árbol generador tal que su peso es mínimo entre todos los árboles generadores del grafo. Consideramos la siguiente estrategia golosa (algoritmo de Prim):

   1\. Inicializar $V_{nuevos} = \{x\}$ donde $x$ es un nodo arbitrario, $E_{nuevos} = \{\}$.
   2\. Repetir hasta que $V_{nuevos} = V$.
   3\.     Tomar un eje $(u, v)$ con peso mínimo tal que $u$ esté en $V_{nuevos}$ y $v$ no.
   4\.     Agregar $v$ a $V_{nuevos}$ y $(u, v)$ a $E_{nuevos}$.
   5\. Devolver $E_{nuevos}$, serán las aristas de un árbol generador mínimo.

   - Convencerse de que este algoritmo devuelve un árbol.
   - Demuestre que el árbol devuelto es un árbol generador mínimo (es decir, la estrategia es óptima).