



PROGRAMACIÓN DINÁMICA [10 puntos] Joffrey I Rey de los Andalos, Lord protector del Reino, se dispone a expandir su corona. Para ello tiene en el punto de mira una lista de R reinos para destruir y hacerse con sus riquezas. Se sabe que cada uno de esos reinos ofrecerá una determinada resistencia. Dicha resistencia está recogida en el vector $\text{Res}[1..R]$, donde $\text{Res}[k]$, con $1 \leq k \leq R$, es la resistencia que ofrecerá el reino k . Se sabe que esa resistencia puede ser vencida con un número igual de soldados. Cada reino le aportaría al Rey Joffrey un determinado enriquecimiento gracias a los recursos de los que dispone: oro, fuego Valyrio, armas, ... y este enriquecimiento está cuantificado y almacenado en $\text{Enri}[1..R]$ donde $\text{Enri}[k]$, con $1 \leq k \leq R$, es el enriquecimiento que se conseguirá al derrotar al reino k .

Como todos sabemos el Rey Joffrey no tiene inteligencia suficiente para solucionar el problema, así que ha pedido ayuda a su tío Tyron Lannister, el cual debe diseñar una estrategia óptima para la conquista sabiendo que no dispone de suficientes soldados en el reino para poder destruir los R reinos simultáneamente.

Con todo ello, Tyron te traslada a ti el diseño de un algoritmo basado en Programación Dinámica que proporcione la mejor estrategia para una conquista simultánea de los reinos objetivo, indicando qué reinos deberán ser destruidos para conseguir el mayor enriquecimiento sin utilizar más soldados de los disponibles, siendo éstos, S .

Se pide responder con claridad y concisión a las siguientes cuestiones:

- Secuencia de decisiones: tamaño y significado de la decisión i -ésima (5%)
- Función Objetivo (10%)
- Restricciones (10%)
- Demostración del principio de optimalidad (15%)
- Ecuación recursiva y primera llamada a la función (30%)
- Árbol de llamadas para un ejemplo (10%)
- Explicar de manera concisa y clara: tipo de estructuras de almacenamiento elegidas, sus dimensiones, cómo se rellenan y cómo se obtiene la solución (20%)