

**2. Diseñar un algoritmo eficiente que, dado un digrafo  $G$  con pesos no negativos, dos vértices  $s$  y  $t$  y una cota  $c$ , determine una arista de peso máximo de entre aquellas que se encuentran en algún recorrido de  $s$  a  $t$  cuyo peso (del recorrido, no de la arista) sea a lo sumo  $c$ . Demostrar que el algoritmo propuesto es correcto.**

Uso Dijkstra desde  $s$ , luego Dijkstra desde  $t$  sobre el grafo transpuesto.

Ahora definimos el conjunto  $A := \{(v, w) \in E(G) : d(s, v) + c(v, w) + d(w, t) \leq c\}$ . Nuestro resultado es el máximo de ese conjunto.

**Correctitud::**

Demostremos que si una arista  $(v, w) \notin A$ , entonces ningún recorrido que haga puede cumplir la cota  $c$ . En particular, como  $d(s, v) + c(v, w) + d(w, t)$  es el mínimo recorrido que va de  $s$  a  $t$  pasando por  $c(v, w)$ , ya que  $d(s, w)$  es la mínima distancia hasta un extremo de la arista y  $d(w, t)$  es la mínima al otro extremo, si este tiene un costo mayor a  $c$ , se sigue que todos los demás lo tendrán. Luego, el conjunto  $A$  define correctamente nuestras aristas candidatas.