

Cognome:..... Nome:..... Matr.:.....

**Esercizio 1 [11 punti]** Si consideri un grafo non orientato e connesso  $G = (V, E)$  con pesi positivi sugli archi. Sia  $S$  un albero dei cammini minimi (*shortest path tree*) di  $G$  con sorgente  $r \in V$ . E sia  $M$  un minimo albero di ricoprente (*minimum spanning tree*) di  $G$ .

1. Dire, per ognuna delle affermazioni seguenti, se è vera o falsa:

- In generale  $S$  ed  $M$  possono essere alberi diversi, ma quando tutti i pesi di  $G$  hanno valore 1 allora è sempre vero che  $S = M$ .
- Se  $G$  è non pesato, ovvero tutti i pesi di  $G$  sono 1, allora  $S$  è *anche* un minimum spanning tree di  $G$ .
- Per ogni nodo  $v$ , il cammino da  $r$  a  $v$  in  $S$  ha lunghezza minore o uguale di quella del cammino da  $r$  a  $v$  in  $M$ .
- Sia  $f$  un arco di  $G$  ma non di  $S$ . Allora  $f$  è l'arco più pesante del ciclo che si forma quando si aggiunge  $f$  ad  $S$ .
- Se  $G$  non ha cicli, allora  $S = M$ .

2. Si consideri la seguente affermazione: *Esistono grafi  $G$  per cui il peso totale di  $S$  è maggiore di due volte il peso totale di  $M$ , dove il peso totale di un albero  $T$  è la somma dei pesi degli archi di  $T$ .* Dire se l'affermazione è vera o falsa motivando la risposta. (Max 5 righe.)

**Esercizio 2 [11 punti]** Si consideri il problema dell'*Interval Scheduling (IS)*.

A. Si definisca formalmente IS. (Max 5 righe.)

B. Si definisca il criterio di ordinamento degli intervalli che porta all'algoritmo greedy corretto, ovvero l'algoritmo greedy che trova sempre una soluzione ottima del problema. (Max 2 righe.)

C. Si dimostri a grandi linee perché l'algoritmo del punto (B) trova sempre una soluzione ottima del problema. (Max 10 righe.)

**Esercizio 3 [11 punti]** Ti è data una sequenza di  $n$  interi  $s_1, s_2, \dots, s_n$  e un intero target  $N$ . Fra ogni coppia di numeri  $s_i$  e  $s_{i+1}$  devi mettere in operatore preso dall'insieme  $\{+, \times, \pm\}$ , che sono l'operatore di somma (+), moltiplicazione  $\times$ , e di skip ( $\pm$ ). L'operatore  $\pm$  ha la seguente semantica:  $a \pm b = a + b - b = a$ . Gli operatori vengono valutati da sinistra a destra e il tuo obiettivo è capire se puoi ottenere il valore  $N$ .

Per esempio, per la sequenza  $s_1 = 1, s_2 = 2, s_3 = 30, s_4 = 8$  è possibile ottenere  $N = 10$ :

$$1 \times 2 \pm 30 + 8 = (1 \times 2) \pm 30 + 8 = 2 \pm 30 + 8 = (2 \pm 30) + 8 = 2 + 8 = 10$$

mentre non è possibile ottenere  $N = 15$ .

Progettate un algoritmo di programmazione dinamica che prende in input la sequenza e l'intero target  $N$  e restituisce **true** se è possibile ottenere  $N$ , **false** altrimenti. Si discuta la complessità temporale dell'algoritmo proposto.