

Algoritmi e Strutture Dati – 02/02/15

Esercizio 1 – Punti ≥ 6 (Parte A)

Trovare limiti superiori e inferiori per la seguente equazione di ricorrenza, utilizzando il metodo di sostituzione (detto anche per tentativi)

$$T(n) = \begin{cases} T\left(\left\lfloor \frac{n}{\sqrt{2}} \right\rfloor\right) + T\left(\left\lfloor \frac{n}{\sqrt{4}} \right\rfloor\right) + T\left(\left\lfloor \frac{n}{\sqrt{8}} \right\rfloor\right) + T\left(\left\lfloor \frac{n}{\sqrt{16}} \right\rfloor\right) + n^2 & n > 1 \\ 1 & n \leq 1 \end{cases}$$

Esercizio 2 – Punti ≥ 6 (Parte A)

Scrivere un algoritmo che dato un grafo diretto G trova il numero di vertici da cui è possibile raggiungere tutti gli altri vertici del grafo.

Discutere informalmente la correttezza della soluzione proposta e calcolare la complessità computazionale.

Esercizio 3 – Punti ≥ 9 (Parte B)

Siano dati n cubi. Su ogni faccia del cubo è presente una lettera dell'alfabeto. Ogni cubo può essere descritto da una stringa di 6 caratteri (e.g. "ABCDEF"), che rappresenta le 6 facce del cubo.

Sia data una parola costituita da $t \leq n$ caratteri; il vostro compito è descrivere un algoritmo che sia in grado di dire se è possibile comporre la parola utilizzando t degli n cubi, scegliendo *una* faccia per ognuno di essi.

Discutere informalmente la correttezza della soluzione proposta e calcolare la complessità computazionale.

Ad esempio, supponete di avere i seguenti cubi: "ABCDEF", "GHIJKL", "AABBCC", "ISTUVW" e di voler comporre la parola IDEA.



Esercizio 4 – Punti ≥ 12 (Parte B)

Per comprarvi l'ultimo gadget tecnologico, avete pazientemente risparmiato inserendo monete in un salvadanaio. Purtroppo, non avete tenuto i conti, e non sapete quanti soldi ci sono dentro. È facile ottenere il valore totale rompendo il salvadanaio, ma sarebbe un peccato romperlo senza essere sicuri che ci siano abbastanza soldi per il vostro gadget.

Fortunatamente, siete informatici e avete a disposizione le seguenti informazioni: il peso totale T delle monete contenute nel salvadanaio, più il vettore dei pesi $p[1 \dots n]$ e dei valori $v[1 \dots n]$, dove $p[i]$ è il peso in grammi e $v[i]$ è il valore in centesimi dell' i -esimo tipo di moneta fra gli n tipi di monete prodotti nel vostro stato.

Scrivere un algoritmo che restituisca il *minimo* valore in centesimi che può essere contenuto nel salvadanaio.

Discutere informalmente la correttezza della soluzione proposta e calcolare la complessità computazionale.

Ad esempio, supponete che il peso totale sia 50 grammi, e le monete a disposizione siano quella da 200 centesimi che pesa 50 grammi, e quella da 50 centesimi che pesa 10 grammi. È possibile ottenere i 50 grammi del peso totale con una singola moneta da 200 centesimi, o con 5 da 50 centesimi per un totale di 250 centesimi. Quindi il valore da restituire è 200, che è il minimo fra i due totali.