

## ***Algoritmi e Strutture Dati - Parte 1 - 25/07/2018***

**Esercizio -1** Iscriversi allo scritto entro la scadenza. In caso di inadempienza, -1 al voto finale.

**Esercizio 0** Scrivere correttamente nome, cognome, numero di matricola, riga e colonna su tutti i fogli consegnati. Consegnare foglio A4 e foglio protocollo di bella. In caso di inadempienza, -1 al voto finale.

**Esercizio A1 – Punti  $\geq 6$**

Si consideri la seguente equazione di ricorrenza, parametrizzata rispetto al valore  $k$

$$T_k(n) = \begin{cases} k^2 T_k(n/k) + n^{k/2} & n > 1 \\ 1 & n \leq 1 \end{cases}$$

Si supponga esistano tre algoritmi, con complessità  $T_2(n)$ ,  $T_3(n)$  e  $T_4(n)$ . Quale algoritmo scartereste? Giustificare la risposta.

PS Suvvia, non vi serve la calcolatrice.

**Esercizio A2 – Punti  $\geq 10$**

Nel gioco dello Shanghai o Mikado, un insieme di  $n$  bastoncini vengono lanciati su un tavolo, con il risultato che i bastoncini si sovrastano l'uno con l'altro. Un bastoncino può essere rimosso se nessun altro bastoncino lo sovrasta. Una volta rimosso, è possibile che i bastoncini che erano sovrastati da esso possano essere rimossi. E' anche possibile tuttavia che ad un certo punto nessun bastoncino possa essere rimosso, in quanto sovrastato da altri bastoncini.

L'input è dato da due vettori  $X$  e  $Y$ , entrambi di dimensione  $m$ , contenenti numeri da 1 a  $n$ . I vettori vanno interpretati in questo modo: per ogni indice  $i$ , il bastoncino  $X[i]$  sovrasta il bastoncino  $Y[i]$ .

Scrivere un algoritmo che prende in input i vettori  $X$ ,  $Y$  oltre alle dimensioni  $n$  ed  $m$ , e restituisce **true** se e solo se è possibile rimuovere tutti i bastoncini presenti, **false** altrimenti.



**Esercizio A3 – Punti  $\geq 14$**

Si consideri un griglia quadrata contenente  $n \times n$  celle. Ogni cella è colorata con uno di tre colori (1,2,3). Supponete di partire da una cella di colore 1. Ad ogni passo potete muovervi di una casella in alto, in basso, a destra o a sinistra. L'obiettivo è raggiungere una cella con colore 3.

Scrivere un algoritmo che prende in input una griglia rappresentata da una matrice di interi e restituisce il numero minimo di passi *necessari* per raggiungere una qualsiasi casella di colore 3 a partire da una qualsiasi casella di colore 1. Per semplicità, supponete che nella griglia sia presente almeno una cella di colore 1 e almeno una cella di colore 3.

Discutere correttezza e complessità dell'algoritmo proposto. Ad esempio, si consideri la matrice seguente:

1	2	2	3
2	1	2	3
2	2	2	<b>3</b>
3	2	<b>1</b>	2

La risposta da dare è 2, perché non esistono celle 1 e 3 adiacenti ma esistono percorsi formati da due passi (come quello evidenziato in grassetto, che però non è l'unico).

*Algoritmi e Strutture Dati - Parte 2 - 25/07/2018*

**Esercizio -1** Iscriversi allo scritto entro la scadenza. In caso di inadempienza, -1 al voto finale.

**Esercizio 0** Scrivere correttamente nome, cognome, numero di matricola, riga e colonna su tutti i fogli consegnati. Consegnare foglio A4 e foglio protocollo di bella. In caso di inadempienza, -1 al voto finale.

## Esercizio B1 – Punti $\geq 8$

Scrivere un algoritmo che prende in input un grafo  $G = (V, E)$ , un nodo  $s \in V$  e un intero  $k$  e stampa tutti i cammini contenuti nel grafo tali che:

- partano da  $s$ ,
  - abbiano lunghezza esattamente  $k$ ,
  - siano semplici (senza nodi ripetuti).

Discutere correttezza e complessità dell'algoritmo proposto.

## Esercizio B2 – Punti $\geq 10$

Tramite un sofisticato algoritmo di Deep Learning, siete ora in grado di predire l'andamento del mercato azionario per i prossimi  $n$  giorni. In particolare, vi concentrate sulle azioni di una singola azienda.

Ogni giorno, potete (1) comprare una singola azione al prezzo corrente, oppure (2) vendere tutte le azioni che avete al prezzo corrente, oppure (3) non fare nulla.

Scrivere un algoritmo che dato un vettore di interi positivi  $V$  contenente il prezzo dell'azione nei prossimi  $n$  giorni, restituisca il profitto massimo che potete ottenere comprando e vendendo azioni. Il profitto è definito dalla somma di quanto guadagnate vendendo azioni meno la somma di quanto spendete comprando azioni. Supponete di aver un capitale iniziale sufficiente per comprare tutte le azioni che vi servono.

Discutere correttezza e complessità dell'algoritmo proposto.

Ad esempio:

- se  $V = [1, 7, 3, 6]$ , vi conviene comprare un'azione nel giorno 1, venderla il giorno 2, comprarne una il giorno 3, venderne una il giorno 4, per un profitto totale di  $(7 + 6) - (3 + 1) = 9$ .
  - se  $V = [7, 1, 2, 5, 3, 1]$ , vi conviene comprare un'azione in ognuno dei giorni 2, 3 e venderle entrambe il giorno 4, ottenendo un profitto di  $(5 \cdot 2) - (1 + 2) = 7$ .

### Esercizio B3 – Punti $\geq$ 12

Avete dato un'occhiata al compito precedente? No? Male!

Volevo dare un problema simile a Somma Massimale, e per fare un esempio ho copiato e incollato una lista di numeri da un altro sito. Come a volte capita, gli spazi non stati copiati e questo è il risultato: "1151832175512". Non trovavo più il sito originale, mi ricordavo solo che la somma era pari a 141.

Nel compito precedente, volevo stampare tutti i modi possibili in cui era possibile suddividere la stringa in una somma di numeri. Questa volta, voglio *contare* tutti i modi possibili in cui è possibile suddividere l'input in una somma di numeri, ottenendo esattamente il valore che mi ricordo.

Scrivere un algoritmo che prenda una stringa di numeri di lunghezza  $n$  (rappresentata da un vettore di interi compresi fra 1 e 9 – "0" escluso) e un intero  $k \leq 100n$ , e restituisca il numero di modi in cui è possibile rappresentare la stringa come sommatoria di interi positivi il cui valore è pari a  $k$ .

Supponete di avere a disposizione una funzione  $\text{value}(V, i, j)$  che restituisce il valore intero corrispondente alle cifre comprese fra le posizioni  $i$  e  $j$  in  $V$ . Ad esempio,  $\text{value}("54321", 2, 4)$  restituisce 432. Questa funzione opera in tempo  $O(n)$ ; sarebbe possibile evitare questo costo, ma non importa.

Discutere correttezza e complessità dell'algoritmo proposto.  
Sorprendentemente, esistono 29 modi per ottenere 141 da 1151832175512;

$$\begin{aligned}
& 11 + 5 + 1 + 83 + 21 + 7 + 5 + 5 + 1 + 2 = 141 \\
& 1 + 15 + 1 + 83 + 21 + 7 + 5 + 5 + 1 + 2 = 141 \\
& 11 + 5 + 1 + 8 + 32 + 1 + 75 + 5 + 1 + 2 = 141 \\
& 1 + 15 + 1 + 8 + 32 + 1 + 75 + 5 + 1 + 2 = 141 \\
& 1 + 1 + 5 + 18 + 32 + 1 + 75 + 5 + 1 + 2 = 141 \\
& 11 + 5 + 18 + 3 + 21 + 75 + 5 + 1 + 2 = 141 \\
& 1 + 15 + 18 + 3 + 21 + 75 + 5 + 1 + 2 = 141 \\
& 11 + 51 + 8 + 3 + 2 + 1 + 7 + 55 + 1 + 2 = 141 \\
& 1 + 1 + 51 + 8 + 3 + 2 + 17 + 55 + 1 + 2 = 141 \\
& 11 + 5 + 18 + 32 + 17 + 55 + 1 + 2 = 141 \\
& 1 + 15 + 18 + 32 + 17 + 55 + 1 + 2 = 141 \\
& 11 + 51 + 8 + 3 + 2 + 1 + 7 + 5 + 51 + 2 = 141 \\
& 1 + 1 + 51 + 8 + 3 + 2 + 17 + 5 + 51 + 2 = 141 \\
& 11 + 5 + 18 + 32 + 17 + 5 + 51 + 2 = 141 \\
& 1 + 15 + 18 + 32 + 17 + 5 + 51 + 2 = 141 \\
& 1 + 1 + 5 + 1 + 83 + 21 + 7 + 5 + 5 + 12 = 141 \\
& 11 + 5 + 1 + 83 + 2 + 17 + 5 + 5 + 12 = 141 \\
& 1 + 15 + 1 + 83 + 2 + 17 + 5 + 5 + 12 = 141 \\
& 11 + 51 + 8 + 32 + 17 + 5 + 5 + 12 = 141 \\
& 1 + 1 + 5 + 1 + 8 + 32 + 1 + 7 + 5 + 5 + 12 = 141 \\
& 11 + 5 + 1 + 8 + 3 + 21 + 75 + 5 + 12 = 141 \\
& 1 + 15 + 1 + 8 + 3 + 21 + 75 + 5 + 12 = 141 \\
& 1 + 1 + 5 + 18 + 3 + 21 + 75 + 5 + 12 = 141 \\
& 1 + 1 + 51 + 8 + 3 + 2 + 1 + 7 + 55 + 12 = 141 \\
& 11 + 5 + 18 + 32 + 1 + 7 + 55 + 12 = 141 \\
& 1 + 15 + 18 + 32 + 1 + 7 + 55 + 12 = 141 \\
& 11 + 5 + 1 + 8 + 32 + 17 + 55 + 12 = 141 \\
& 1 + 15 + 1 + 8 + 32 + 17 + 55 + 12 = 141 \\
& 1 + 1 + 5 + 18 + 32 + 17 + 55 + 12 = 141
\end{aligned}$$