# Algoritmi e Strutture Dati - Parte A - 07/02/2022

Esercizio A1 - Ricorrenza - Punti 9

Esercizio A1 – Ricorrenza – Punti  $\geq 9$ 

Trovare i limiti superiore e inferiore più stretti possibili per la seguente equazione di ricorrenza:

$$T(n) = \begin{cases} 1 & n < 27 \\ 12T(\lfloor n/8 \rfloor) + 18T(\lfloor n/27 \rfloor) + n\sqrt[3]{n} & n \ge 27 \end{cases}$$

## Esercizio A2 – Gazprom – Punti $\geq 9$

Si consideri una rete di n depositi di gas, collegati fra loro da m condutture. Ogni deposito u ha una capacità positiva C[u].

Ogni deposito ha una valvola di ingresso, in cui è possibile pompare del gas. Se pompate gas in un deposito, questo si riempie fino a colmare la sua capacità e poi il gas inizia a fluire per le condutture, riempiendo i depositi a esso collegati. Questi si riempiono fino alla loro capacità e poi il gas inizia a fluire nei depositi a essi collegati non ancora riempiti, e così via. La quantità di gas che rimane nelle condutture tra un deposito e l'altro è zero. <sup>1</sup>

Scrivere un algoritmo

che prende in input un grafo non orientato G=(V,E) (dove V rappresenta l'insieme dei depositi e E rappresenta l'insieme delle condutture) e un vettore di capacità positive C, e restituisce la quantità massima di gas che può essere pompata nella rete utilizzando la valvola di ingresso di uno e un solo deposito.

Discutere informalmente la correttezza dell'algoritmo e la sua complessità computazionale.

### Esercizio A3 – Alberi pasciuti – Punti $\geq 12$

Un albero binario si dice *pasciuto* se ogni livello  $\ell$  dell'albero contiene almeno  $2^{\ell-1}$  nodi.

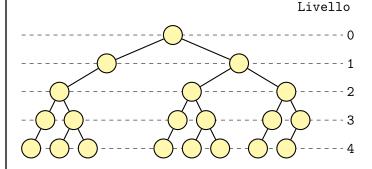
Scrivere un algoritmo

#### **boolean** is Plump(TREE T)

che prende in input un albero binario T non vuoto e restituisce **true** se T è un albero pasciuto, **false** altrimenti.

Discutere informalmente la correttezza dell'algoritmo e la sua complessità computazionale.

Per esempio, l'albero seguente è un albero binario pasciuto. Se cancellassimo uno qualunque dei nodi di livello 4, non sarebbe più pasciuto.



<sup>&</sup>lt;sup>1</sup>La fisica dei gas non si comporta così, lo so.

## Algoritmi e Strutture Dati - Parte B - 07/02/2022

### Esercizio B1 – Bit consecutivi – Punti > 8

Scrivere un algoritmo;:

binary(int 
$$n$$
, int  $n_0$ , int  $n_1$ )

che prende in input un intero positivo n e due interi positivi  $n_0$ ,  $n_1$  e stampa tutte le stringhe binarie di n cifre che contengono al massimo  $n_0$  bit 0 consecutivi e al massimo  $n_1$  bit 1 consecutivi.

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

Per esempio, con n = 6,  $n_0 = 2$ ,  $n_1 = 3$ , il programma deve stampare queste stringhe (non necessariamente in quest'ordine):

#### Esercizio B2 – Discordville – Punti > 11

Discordville è un villaggio particolare, composto da n case, numerate da 1 a n lungo una singola strada. A Discordville ogni casa è in disaccordo con un certo numero di case che si trovano prima di essa lungo la strada. Discordville vuole organizzare una sagra e ha lanciato una raccolta fondi che è vostro compito organizzare.

Ogni abitante i è in grado di donare una quantità D[i], ma non intende partecipare a una raccolta fondi a cui partecipano le precedenti N[i] case, ovvero le case in posizione  $i-1, i-2, \ldots, i-N[i]$ .

Scrivere un algoritmo;:

int discordville(int[] 
$$D$$
, int[]  $N$ , int  $n$ )

che prende in input il vettore D contenente le donazioni (interi non negativi), il vettore N contenente il numero di case in disaccordo (interi non negativi) e il numero totale di case n, e restituisce il totale massimo di donazioni che possono essere raccolte.

Discutere informalmente la correttezza della soluzione proposta e calcolare la complessità computazionale. Per esempio, si consideri:

- D = [3, 4, 5, 7, 1]
- N = [0, 1, 1, 3, 1]

Se si prende il valore 7, non è possibile prende il valore 1 successivo (perché la casa in posizione 5 non vuole donare insieme alla casa in posizione 4) e nemmeno le tre case precedenti; conviene prendere 1, 5 e 3, per un totale di 9.

### Esercizio B3 – Math ways – Punti > 11

Scrivere un algoritmo

int mathWays(int 
$$n$$
, int  $k$ )

che restituisce il numero di modi distinti per ottenere il valore n a partire dal valore k utilizzando le seguenti operazioni:

- aggiungere 1 al numero corrente;
- raddoppiare il numero corrente;
- elevare il numero corrente al quadrato.

Per semplicità, si assuma che  $n \geq 3$  e  $k \geq 3$ .

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

Per esempio, è possibile ottenere n = 10 a partire da k = 3 in cinque modi distinti:

- $3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10$
- $3 \rightarrow 4 \rightarrow 5 \rightarrow 10$
- $3 \rightarrow 4 \rightarrow 8 \rightarrow 9 \rightarrow 10$
- $3 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10$
- $3 \rightarrow 9 \rightarrow 10$

e quindi l'algoritmo deve restituire 5.