Cognome: **Nome:** **# Matricola:** **Riga:** **Col:**

Algoritmi e Strutture Dati – 10/01/2017

Esercizio 1 – Punti ≥ 6 (Parte B)

Dato un vettore V di interi appartenenti all'insieme $\{0, 1\}$, si scriva un algoritmo che restituisca la lunghezza del più lungo sottovettore contiguo formato da k valori 0 seguiti da k valori 1. Si noti che è possibile che esistano altri valori 0 prima del sottovettore così individuato, oppure altri valori 1 dopo il sottovettore, ma non è possibile che si verifichino entrambe le estensioni, altrimenti il sottovettore non sarebbe massimale.

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

Esempio: per l'input $\texttt{001111111}\underline{\texttt{000111}}\texttt{10000}$, l'algoritmo deve restituire 6.

Esercizio 2 – Punti ≥ 7 (Parte B)

Una stringa P è una supersequenza di una stringa T se T è una sottosequenza di P. Scrivere un algoritmo che restituisce la lunghezza della *supersequenza comune minimale* di due stringhe P, T, ovvero la più piccola supersequenza di entrambe le stringhe.

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

Esempio: L'unica supersequenza comune minimale di `AB` e `BC` è `ABC`, e la sua lunghezza è pari a 3.

Esempio: Esistono due supersequenze comuni minimali di `DAB` e `DCB`, ovvero `DACB` e `DCAB`, e la loro lunghezza è pari a 4.

Esercizio 3 – Punti ≥ 8 (Parte B)

Data una rete di flusso $G = (V, E, s, t, c)$ con capacità intere, un arco (u, v) è detto *upward-critical* se aumentando la sua capacità di 1, il flusso massimo della rete aumenta; un arco (u, v) è detto *downward-critical* se diminuendo la sua capacità di 1, il flusso massimo della rete diminuisce.

1. Mostrare una rete di flusso (con sorgente, pozzo, capacità) in cui esiste almeno un arco upward-critical, se possibile

2. Mostrare una rete di flusso (con sorgente, pozzo, capacità) in cui non esiste alcun arco upward-critical, se possibile

3. Mostrare una rete di flusso (con sorgente, pozzo, capacità) in cui esiste almeno un arco downward-critical, se possibile

4. Mostrare una rete di flusso (con sorgente, pozzo, capacità) in cui non esiste alcun arco downward-critical, se possibile

Esercizio 4 – Punti ≥ 10 (Parte B)

Siano dati n file. Il file i-esimo ha lunghezza ℓ_i e una probalità di accesso p_i.
I file devono essere memorizzati in un nastro ad accesso sequenziale, secondo una qualche permutazione $\pi = \pi(1)\pi(2)\ldots\pi(n)$ dell'insieme degli indici $\{1, \ldots, n\}$. Il costo c_i richiesto per leggere l'i-esimo file della permutazione è pari alla somma delle lunghezze dei primi i file nella permutazione:

$$c_i = \sum_{j=1}^{i} \ell_{\pi(j)}$$

Il costo atteso di lettura è pari a:

$$C = \sum_{i=1}^{n} p_{\pi(i)} \cdot c_i$$

ovvero la sommatoria del prodotto dei costi di lettura di un file per la probabilità che tale file un certo file venga letto, per ogni file
ovvero la media dei prodotti delle frequenze di accesso per il costo di lettura per ogni file della permutazione.
Vi è stato richiesto di scrivere un algoritmo che minimizzi il costo medio di lettura. Considerate i seguenti algoritmi greedy:

- Si ordinino i file per lunghezza ℓ_i crescente

- Si ordinino i file per probabilità di accesso p_i decrescente

Per ognuno degli algoritmi proposti, si provi che restituiscono il minor tempo medio di lettura, oppure si individui un controesempio. Nel caso entrambi gli algoritmi non siano ottimali, si suggerisca un nuovo algoritmo che risolva il problema, dimostrandone se possibile la correttezza.