

Algoritmi e Strutture Dati

a.a. 2012/13

Compito del 06/09/2013

Cognome: _____

Nome: _____

Matricola: _____

E-mail: _____

Parte I

(30 minuti; ogni esercizio vale 2 punti)

1. In una tabella Hash di $m = 17$ posizioni, inizialmente vuota, devono essere inserite le seguenti chiavi numeriche nell'ordine indicato:

52, 1, 92, 37

La tabella è a indirizzamento aperto e la scansione è eseguita per doppio Hashing:

$$h(k, i) = (k \bmod m + i * 2^{k \bmod 5}) \bmod m$$

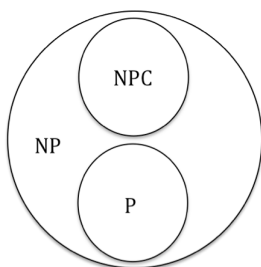
Indicare per ogni chiave le posizioni scandite nella tabella e la posizione finale dove viene allocata.

2. Si definiscano le relazioni O , Ω , Θ e, utilizzando le definizioni date e nient'altro, si mostri che valgono le seguenti proprietà:

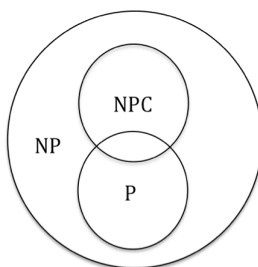
a) $f(n) = O(g(n))$ se e solo se $g(n) = \Omega(f(n))$

b) $f(n) = \Theta(g(n))$ se e solo se $g(n) = \Theta(f(n))$

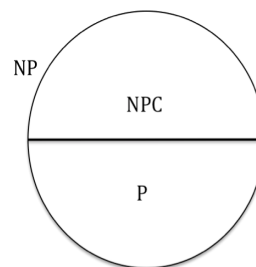
3. Si definiscano le classi P, NP, NPC e si stabilisca, giustificando formalmente la risposta, quale delle seguenti relazioni è ritenuta vera (o verosimile):



(a)



(b)



(c)

Algoritmi e Strutture Dati

a.a. 2012/13

Compito del 06/09/2013

Cognome: _____

Nome: _____

Matricola: _____

E-mail: _____

Parte II

(2.5 ore; ogni esercizio vale 6 punti)

1. Sia v un array di n interi distinti, tale che esiste una posizione j , $0 \leq j < n$, per cui:
 - a. gli elementi nel sottoarray $v[0, j]$ sono ordinati in ordine decrescente;
 - b. gli elementi in $v[j+1, n-1]$ sono in ordine crescente;
 - c. $v[j] < v[j+1]$, se $j < n-1$

Scrivere un algoritmo di tipo *divide-et-impera* **efficiente** che restituisca la posizione j . Calcolare la complessità al caso peggior dell'algoritmo indicando, e risolvendo, la corrispondente relazione di ricorrenza.

Per l'esame da **12 CFU**, deve essere fornita **una funzione C**.

Per l'esame da **9 CFU**, è sufficiente specificare lo pseudocodice.

2. Dato un albero di ricerca T , progettare un algoritmo **efficiente** che restituisca il numero di elementi che occorrono una sola volta e analizzarne la complessità.
 - a. Non si possono usare strutture ausiliarie di dimensione **$O(n)$** dove n è il numero dei nodi dell'albero.
 - b. Devono essere definite esplicitamente eventuali funzioni/procedure ausiliarie. Si consideri la rappresentazione dell'albero binario di ricerca che utilizza i campi **left**, **right**, **p** e **key**.
3. Sia $G = (V, E)$ un grafo orientato e pesato e sia $s \in V$ un vertice "sorgente". Supponendo che G sia stato inizializzato con INIT-SINGLE-SOURCE(G, s), si dimostri che se una qualsiasi sequenza di passi di rilassamento assegna a $\pi[s]$ un valore diverso da NIL, allora G contiene un ciclo di peso negativo raggiungibile dalla sorgente s . (Suggerimento: si utilizzino le definizioni delle funzioni citate e le proprietà elementari della funzione RELAX).
4. Si vuole costruire una rete stradale che colleghi cinque città (A-E), minimizzando i costi complessivi di realizzazione. I costi per la costruzione di una strada tra due città sono sintetizzati nella seguente tabella (dove $+\infty$ significa che la strada è irrealizzabile):

	A	B	C	D	E
A	0	3	5	11	9
B	3	0	3	9	8
C	5	3	0	$+\infty$	10
D	11	9	$+\infty$	0	7
E	9	8	10	7	0

Si formuli il problema dato in termini di un problema di ottimizzazione su grafi, e si descriva un algoritmo per la sua soluzione discutendone correttezza e complessità. Infine, si simuli accuratamente l'algoritmo presentato per determinare una soluzione del problema.