

Seconda Provetta

ASD1 2002-2003

Exercise 1 Dimostrare le seguenti proprietà relative ai numeri di Fibonacci.

1. $F_{k+2} = 1 + \sum_{i=0}^k F_i$;
2. $F_{k+2} \geq \phi^k$, dove $\phi := \frac{1+\sqrt{5}}{2}$ indica la sezione aurea.

Exercise 2 Si consideri una foresta di alberi gestita da una famiglia di guardiaboschi. All'inizio la foresta non contiene alcun albero. I guardiaboschi eseguono, in sequenze arbitrarie, operazioni delle seguenti 3 tipologie.

1. **impianto:** impiantare un nuovo albero fatto di un solo nodo senza discendenza;
2. **diversificazione:** individuare due alberi T_1 e T_2 della stessa specie (ossia con radici aventi lo stesso numero di figli), sradicare T_1 ed innestare la radice r_1 come figlio diretto della radice di T_2 ;
3. **potatura:** la potatura è un'operazione ricorsiva. Dato un nodo v in un albero, indichiamo con $\pi(v)$ il papà di v , ossia il nodo in cui v risulta innestato come figlio. Potar via un nodo v significa staccare da $\pi(v)$ il sottoalbero radicato in v ed innestarlo direttamente nel terreno, ed inoltre, se $\pi(v)$ si era già visto potar via un figlio precedentemente a v , da quando si trova innestato come figlio del suo papà $\pi(\pi(v))$, allora la potatura si applica ricorsivamente a $\pi(v)$.

Con riferimento al precedente esercizio, sapresti dimostrare che in un qualsiasi momento, ed ove v sia un qualsiasi nodo di albero della nostra foresta con, diciamo, k figli, allora la cardinalità della discendenza di v , ossia il numero di nodi del sottoalbero radicato in v , è almeno ϕ^k .

Exercise 3 Sapresti dare un pseudocodice od una descrizione comunque efficace e formale dell'operazione di EstraiMinimo (e Consolida) relativa ad uno Heap di Fibonacci? Sapresti indicare il costo reale nel caso peggiore e sapresti tracciare l'analisi ammortizzata?

Exercise 4 Agli archi di un grafo $G = (V, E)$ sono associate delle lunghezze (valori numerici non negativi). Abbiamo cioè una funzione $l : E \mapsto R_+$. Inoltre viene indicato un nodo sorgente s . Vorremo valutare la distanza $d(s, v)$ per ogni nodo in V , ossia la minima lunghezza di un cammino da s a v . Poggiando su uno heap di servizio (preso come black-box), si dia un pseudocodice per l'algoritmo di Dijkstra che computa le distanze $d(s, v)$ da un dato nodo sorgente s a tutti gli altri nodi, ma anche i cammini ottimi (si utilizzi un array di predecessori $\pi : V \mapsto V \cup \{-1\}$), qualora le lunghezze degli archi siano tutte non negative. Nello pseudocodice, i nodi siano colorati bianco, grigio e nero, secondo il seguente schema:

bianco se non si è ancora individuato un cammino da s al nodo in questione;

grigio se si è già individuato qualche cammino da s al nodo in questione v , ma v non è stato ancora estratto dallo heap di servizio;

nero il nodo in questione è già stato estratto dallo heap di servizio.

Dimostrare poi le seguenti invarianti:

1. ad ogni iterazione, ad uno specifico passo della tua procedura da te indicato, puoi affermare che i nodi grigi sono tutti e soli i nodi nello heap di servizio;
2. sempre, $\lambda(v) \geq d(v)$;
3. quando v è nero, allora $\lambda(v) = d(v)$.

Exercise 5 *Con riferimento al precedente esercizio, sapresti fornire lo pseudocodice di una procedura che, dato un nodo v , e sulla base del vettore dei predecessori π come computato dall'algoritmo di Dijkstra, restituisca un cammino ottimo da s a v in tempo lineare nella lunghezza del cammino stesso? Sapresti condurre l'analisi del running time nel caso peggiore evidenziando inoltre le differenze qualora lo heap sia stato realizzato come Heap Binomiale o come Heap di Fibonacci?*

Exercise 6 *Sapresti dare uno pseudocodice od una descrizione comunque efficace e formale dell'algoritmo RadixSort che usi come algoritmo intermedio il CountingSort preso come black-box? Sapresti indicare il tempo di calcolo nel caso peggiore e migliore? Riesci ad individuare una invariante di ciclo? Se al posto del CountingSort utilizzi un altro algoritmo di ordinamento, l'output resta comunque corretto? (Fornire eventuale controesempio).*