

Algoritmica 2

Appello 2: 02/02/2010

Esercizio 1

[punti 7]

Si deve applicare l'algoritmo randomizzato di Miller e Rabin per stabilire se un numero N è primo con probabilità di errore p prefissata. Ricordiamo che posto $N-1 = 2^w z$ con z dispari, e scelto y arbitrario compreso tra 2 e $N-1$, l'algoritmo impiega i due predicati:

$$P_1. \text{ mcd}(N,y)=1$$

$$P_2. (y^z \bmod N = 1) \text{ OR (esiste } i \text{ intero tra } 0 \text{ e } w-1 \text{ tale che } y^{jz} \bmod N = -1, \text{ con } j = 2^i)$$

P_1 e P_2 sono soddisfatti da ogni numero primo N , e sono soddisfatti da un numero composto per non più di $N/4$ valori di y .

Spiegare in cosa consiste l'algoritmo e qual'è la complessità delle operazioni necessarie.

Esercizio 2

[punti 7]

Supponiamo che H sia una classe di funzioni hash universali tali che $h: U \rightarrow \{0,1,2,\dots,m-1\}$.

- a) Sia S un sottoinsieme di U formato da n chiavi. Si dimostri che, presa a caso una funzione h da H , il numero medio di collisioni generate da h sulle chiavi di S è $< n^2/(2*m)$.
- b) Usare H per progettare uno "schema di fingerprinting" per le chiavi di S tale che, per ogni chiave genera $O(\log n)$ bits, e garantisce in media no collisioni tra le fingerprint delle chiavi di S .
- c) Sia dato un multi-insieme X di cardinalità x , formato da elementi di U (possibilmente ripetuti, dunque). Progettare un algoritmo randomizzato che CONTA il numero di chiavi duplicate di X (non importa che trovi quali siano) in tempo e spazio $O(n)$. L'algoritmo è di tipo Montecarlo e quindi può restituire un conteggio errato, per cui si valuti la sua probabilità di fallimento. [Sugg. Se i numeri da ordinare sono in $O(n^2)$ allora....]

Esercizio 3

[punti 7]

a) Si dia il Suffix Array della stringa ABRACADABRA\$ di lunghezza $m=12$.

b) Si descriva in modo rigoroso ma sintetico come, in generale, usando la ricerca binaria sul suffix array di un testo T , si possano individuare tutte le occorrenze (o dire se non ve ne sono) di un dato pattern P nel testo T . [Questo metodo non deve essere necessariamente il piu' efficiente]

c) Si esemplifichi in particolare l'applicazione del metodo descritto in b) alla ricerca del pattern ABR nel testo ABRACADABRA.

Esercizio 4

[punti 7]

Progettare un algoritmo per il Web-caching di pagine p , che tenga conto del costo $c(p)$ e della dimensione $s(p)$ di ciascuna pagina e che realizzi il principio di località mediante la tecnica della marcatura: si tolga dalla memoria cioè la pagina meno conveniente tra le pagine non marcate.

Esercizio 5

[punti 7]

Dato un array A di N interi, memorizzato in memoria esterna, progettare un algoritmo efficiente per calcolare le somme prefisse, analizzandone la complessità nel modello EMM:

$$p(i) = A[1] + A[2] + \dots + A[i],$$

per tutti i valori $1 \leq i \leq N$. Discutere cosa succede invece se A è una lista di N interi, rispetto alla situazione descritta sopra.