

Monete

Siano date n monete delle quali una falsa. Si supponga che tutte le monete abbiano lo stesso peso, tranne quella falsa che pesa di meno. Si chiede di fornire un algoritmo di complessità $O(\log n)$ che determini la moneta falsa usando una bilancia a due piatti che può confrontare il peso di sottoinsiemi di monete. Mostrare che il problema ha complessità $\Omega(\log n)$ e quindi l'algoritmo è ottimo. Si può generalizzare al caso in cui si sa solo che la moneta falsa ha un peso diverso da quelle vere, ma non è noto se pesi di più o di meno?

Soluzione Lavoriamo nell'ipotesi che la moneta falsa pesi di meno e iniziamo rispondendo alla domanda sulla complessità del problema, ovvero mostrando che il problema ha complessità $\Omega(\log n)$.

Detto S l'insieme di monete, una *pesata* è un'operazione che prende due sottoinsiemi disgiunti di monete S_1 e S_2 e li confronta tramite la bilancia. La pesata sarà maggiormente informativa se i due insiemi scelti contengono lo stesso numero di monete. Più precisamente abbiamo

$$S = S_1 \uplus S_2 \uplus S_3$$

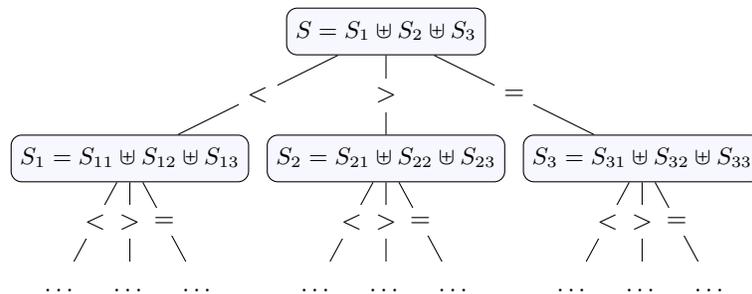
con $|S_1| = |S_2|$ e $S_i \cap S_j = \emptyset$ per $i \neq j$. In questa situazione, se:

- peso $S_1 <$ peso S_2 : la moneta falsa è in S_1 ;
- peso $S_1 =$ peso S_2 : la moneta falsa è in S_3 ;
- peso $S_1 >$ peso S_2 : la moneta falsa è in S_2 .

Quindi ci si riduce a cercare la moneta in uno dei tre sottoinsiemi.

A margine, si ribadisce che considerare sottoinsiemi S_1 e S_2 che contengano un numero diverso di monete non è conveniente. Infatti, supponendo ad esempio che $|S_1| < |S_2|$, se peso $S_1 <$ peso S_2 non si ha alcuna informazione, la pesata è stata inutile!

L'esecuzione di un qualunque algoritmo che evidenzi le operazioni "pesata" può essere rappresentato mediante un albero di decisione del tipo:



Si osserva che, qualunque sia l'algoritmo seguito

- in ciascuno nodo, almeno uno dei sottoinsiemi ha cardinalità maggiore o uguale a

$$(\text{numero totale di monete nel nodo})/3;$$

- le foglie, per portare ad una decisione, devono avere cardinalità 1.

quindi l'albero ha altezza $\geq \log_3 n$.

La complessità (numero di pesate) dell'algoritmo nel caso peggiore è data dal cammino più lungo dalla radice ad una foglia (altezza dell'albero). Infatti, al variare dell'input, essendo le monete indistinguibili, potrò ottenere un'esecuzione corrispondente ad uno qualunque dei cammini.

Dato che questo vale per ogni algoritmo che risolve il problema, $\log_3 n$ è un limite inferiore per la complessità, e dunque il problema ha complessità $\Omega(\log n)$.

Algoritmo: Un algoritmo ottimo si può ottenere dividendo l'insieme corrente di n monete in tre sottoinsiemi $S = S_1 \uplus S_2 \uplus S_3$ con $|S_1| = |S_2| = \lfloor n/3 \rfloor$. Questo porta ad una complessità $O(\log_3 n)$, quindi ottima. Anche una suddivisione con $|S_1| = |S_2| = \lfloor n/2 \rfloor$ porta ad un algoritmo asintoticamente ottimo, dato che la complessità sarà $O(\log_2 n)$ e la base del logaritmo è irrilevante dal punto di vista del limite asintotico.

Se non è noto se la moneta falsa pesa di più o di meno: L'algoritmo precedente può essere facilmente adattato purché $n \geq 3$. Alla prima pesata, si considerano $|S_1| = |S_2| = \lfloor n/3 \rfloor$. Se S_1 pesa di più, significa che S_3 è interamente costituito da monete vere. Dunque, preso un sottoinsieme S'_3 di S_3 con cardinalità $\lfloor n/3 \rfloor$, e confrontatolo con S_1 :

- se S'_3 pesa di meno di S_1 , allora S_1 contiene la moneta falsa e la moneta falsa pesa di più di quelle vere, si continua dunque come nel già trattato (dualizzando);
- se S'_3 e S_1 hanno lo stesso peso, allora S_2 contiene la moneta falsa, che pesa di meno di quelle vere e si continua dunque come nel caso precedente;
- si noti che non può essere che S'_3 pesi di più di S_1 , perché S_1 a sua volta pesa più di S_2 , mentre il peso di S'_3 deve coincidere con quello di S_1 o di S_2 .

Se invece alla prima pesata, S_1 e S_2 hanno lo stesso peso, sono entrambi costituiti da monete vere, posso dunque estrarre da $S_1 \cup S_2$ un sottoinsieme di monete vere S' con la stessa cardinalità di S_3 . Confrontando S' con S_3 capirò se la moneta falsa, necessariamente contenuta in S_3 , pesa di più o di meno di quelle vere. A questo punto si continua come prima.

In ogni caso, aggiungo un'unica pesata all'algoritmo, la cui complessità asintotica non cambia.