

Corso di Laurea in Informatica
Corso di Algoritmi
Esercizi proposti – 11 marzo 2015

1. Blocchi di lunghezza pari

Data una stringa binaria, definiamo *blocco* una sottostringa di simboli uguali consecutivi che non può essere estesa. Scrivere un algoritmo che, dato un intero n stampi tutte le stringhe binarie di lunghezza n che non contengono blocchi di lunghezza pari.

Ad esempio se $n = 5$ l'algoritmo deve stampare:

00000, 00010, 01000, 01010, 01110, 10101, 10001, 11101, 10111, 11111.

La complessità dell'algoritmo deve essere $O(n D(n))$, dove $D(n)$ è il numero di stringhe da stampare.

SOLUZIONE

- Il vettore X contiene le varie soluzioni che vengono poi "stampate" (o comunque trattate)
- i è l'indice del vettore X da assegnare (0 o 1).
- $pari$ è una variabile booleana (valori T o F) che indica se l'ultimo blocco del vettore riempito è di lunghezza pari.
- Chiamata ($n, -, X, 1$) //il valore di $pari$ è irrilevante

Bl-dis($n, pari, X, i$)

```
if (i > n) then if (pari = F) then stampa(X[1], ..., X[n])
                return
if (i = 1) then X[1] ← 0
                Bl-dis (n, F, X, i+1)
                X[1] ← 1
                Bl-dis (n, F, X, i+1)
else if (pari = T) then X[i] ← X[i-1]
                       Bl-dis (n, F, X, i+1)
                       else if (i ≠ n) then X[i] ← X[i-1]    \pari = F
                                           Bl-dis (n, T, X, i+1)
                                           X[i] ← (X[i-1] + 1) mod 2
                                           Bl-dis (n, F, X, i+1)
```

2. Numero di 1 consecutivi

Scrivere un algoritmo di backtracking che, presi in input due interi positivi n e k , con $k \leq n$, stampi tutte le sequenze binarie di lunghezza n che contengono almeno k simboli 1 consecutivi. Ad esempio, per $n = 4$ e $k = 2$ si devono stampare, non necessariamente in questo ordine: 0011, 0110, 1100, 1101, 1011, 1110, 0111, 1111.

3. Stringhe binarie controllate

Fornire un algoritmo di backtracking che, presi in input due interi n e k , con $k \leq n/2$, stampi tutte le stringhe binarie di lunghezza n in cui il numero di 1 è almeno uguale al numero di zero e il numero di zero è almeno k . Ad esempio per $n = 5$ e $k = 2$ si deve stampare, non necessariamente in questo ordine:

00111, 01011, 01101, 01110, 10011, 10101, 10110, 11001, 11010, 11100.

4. Somma di sottoinsieme

Dati un insieme di interi $A = \{a_1, a_2, \dots, a_n\}$ e un intero positivo k , esiste un sottoinsieme S degli indici $\{1, 2, \dots, n\}$ tale che $\sum_{i \in S} a_i = k$?

Scrivere un algoritmo di programmazione dinamica e uno di backtracking per risolvere il problema “Somma di Sottinsieme” e studiarne la complessità. In quali condizioni è necessario usare quello di backtrack?

5. Somma di sottoinsieme limitato

Nel caso dell'esercizio precedente si supponga di cercare un sottoinsieme S di indici tali che $\sum_{i \in S} a_i \leq k$ ma sia il massimo possibile tra le somme di sottoinsiemi di S di valore minore o uguale a k . Scrivere un algoritmo di branch and bound per trovare la soluzione.