IOI ’98: Party Lamps

Luca Foschini

19/02/2002

Il problema

Per illuminare il salone delle IOI ’98 abbiamo a disposizione un set di N lampade colorate numerate da 1 a N. Le lampade sono connesse a 4 pulsanti:

- Pulsante 1: quando viene premuto tutte le lampade cambiano il loro stato: quelle spente si accendono e quelle che sono accese si spengono.
- Pulsante 2: cambia lo stato delle lampade con numero dispari.
- Pulsante 3: cambia lo stato delle lampade con numero pari.
- Pulsante 4: cambia lo stato delle lampade nella forma $3k + 1$ con $k \geq 0$, quindi $1, 4, 7, \ldots$.

C’è un contatore C che registra il numero totali di pressioni di bottoni. Quando la festa comincia tutte le lampade sono accese e il contatore C ha valore zero.

Compito. Ti viene fornito il valore del contatore C e informazioni sullo stato finale di alcune lampade. Ti viene richiesto di scrivere un programma che determini tutte le possibili configurazioni finali delle N lampade in accordo con le informazioni fornite, senza ripetizioni.

La soluzione

Prendiamo in considerazione un numero N di lampade, e notiamo che:

- Conoscendo lo stato della lampada n. 2 possiamo dedurre lo stato di tutte le lampade con numero pari non nella forma $3k + 1$, dato che le lampade che portano un numero con queste proprietà possono essere modificate solo dalla pressione del bottone 3 (e dall’1, ma questo per ora non ci riguarda).
- Conoscendo lo stato della lampada n. 3 possiamo dedurre lo stato di tutte le lampade con numero dispari non nella forma $3k + 1$, dato che le lampade che portano un numero con queste proprietà possono essere modificate solo dalla pressione del bottone 2 (e dall’1).
- Conoscendo lo stato della lampada n. 1 e conoscendo lo stato delle lampade con numero dispari non nella forma $3k + 1$ (ad esempio la n. 3), possiamo dedurre lo stato delle lampade con numero dispari nella forma $3k + 1$ e, allo stesso modo, possiamo dedurre lo stato di tutte le lampade con numero pari nella forma $3k + 1$ conoscendo lo stato delle lampade con numero pari non nella forma $3k + 1$ (ad esempio la n. 2).

La pressione del bottone n. 1, influenzando tutte le lampade allo stesso modo, di fatto è come se non ne influenzasse nessuna (in quanto le relazioni sopra descritte rimangono valide).

Ora sappiamo quindi che la configurazione finale delle N lampade è univocamente identificata dallo stato delle prime 3 lampade.

È inoltre evidente che la lampada n. 7 è nello stesso stato della lampada n. 1, e la n. 8 della n. 2. Lo stato delle N lampade non è altro che una concatenazione ripetuta più volte del pattern lungo 6 caratteri che descrive lo stato delle prime 6 lampade (gli stati delle lampade 4, 5, 6 sono funzione degli stati delle lampade 1, 2, 3, ma non sono identici).

I possibili stati delle prime tre lampade sono $2^3 = 8$; abbiamo quindi ridotto notevolmente le spazio delle soluzioni.

Ora è quindi possibile fare un uso più efficiente delle informazioni parziali fornite in input e considerare l'informazione riguardante la i-esima lampada come se riguardasse la ($i \mod 6$)-esima. Il problema si riduce quindi a un confronto tra tra 8 stringhe di lunghezza 6 caratteri.

Con un po' di pazienza si ricava che le 8 stringhe di bit (configurazioni finali possibili) sono:

1. 000000 = 0 in decimale
2. 111111 = 63
3. 101010 = 42
4. 010101 = 21
5. 011011 = 27
6. 100100 = 36
7. 001110 = 14
8. 110001 = 49

Da questo discende che esisteranno anche solo 8 possibili sequenze di mosse diverse che portano dalla configurazione iniziale (tutte le lampade accese) a una delle 8 finali.

Ma ciò non è sufficiente. Non possiamo ancora dire nulla sul numero di pressioni di bottoni che portano a una delle 8 configurazioni finali partendo dalla configurazione iniziale (anche se si può già intuire qualcosa); serve quindi fare un paio di considerazioni sul dato C.

Notiamo che la operazioni di pressione di bottone (che da ora in poi chiameremo "mosse") sono commutative, ovvero lo stato finale delle lampade dipende solo da quali mosse sono state effettuate, e non dall'ordine in cui le mosse sono state fatte.

È altresì evidente che compiendo due volte la stessa mossa, a partire da una configurazione si ritorna alla configurazione stessa.

Ogni sequenza di mosse, di qualsivoglia lunghezza, può quindi essere ridotta a una lunga non più di 4 mosse (essendo 4 le possibili mosse diverse).

Ogni sequenza di mosse possibili è quindi equivalente a un sottinsieme delle 4 possibili mosse e il numero di possibili sottinisiemi di un insieme di 4 elementi è, come sappiamo, $2^4 = 16$

Si giunge quindi a determinare l'esistenza di solo 16 sequenze di mosse (compresa la mossa nulla, che consiste nella pressione di nessun bottone), ma sfruttando alcune considerazioni sull'equivalenza di alcune sequenze di mosse (premere il bottone 2 e poi il 3 equivale a premere il bottone 1, la pressione dell'1 e del 3 equivale alla pressione del 2, e, viceversa, la pressione dell'1 e del 3 equivale alla pressione del 2) si giunge a ricavare che ogni sequenza lunga un numero arbitrario di mosse è riducibile a una sequenza lunga al più due mosse, per un totale di 8 possibili sequenze che chiameremo sequenze minime.

In particolare:

1. Se C= 1 le possibili sequenze di mosse sono 1 ; 2 ; 3 ; 4.
2. Se C= 2 le possibili sequenze di mosse sono 0 ; 1 ; 2 ; 3 ; 1,4 ; 2,4 ; 3,4 (dove 0 indica la pressione di nessun bottone).
3. Se C= 3 le possibili sequenze di mosse sono tutte quelle minime (e quindi, come dimostrato prima, tutte quelle possibili) quindi: 0 ; 1 ; 2 ; 3 ; 4 ; 1,4 ; 2,4 ; 3,4.

Si giunge alle considerazioni precedenti facendo uso solo di un foglio di carta e un matita.

Un'ultima considerazione ci porta alla certezza matematica del fatto che il problema abbia soluzione: basta osservare che ogni configurazione raggiungibile con k mosse lo è anche con $k + 2t$ mosse, dove t è un intero positivo (basta aggiungere alla sequenza una doppia pressione dello stesso bottone) ma anche $k + 3t$ mosse, (basta aggiungere, per esempio, la sequenza di pressioni 1;2;3).

Ognuna delle 8 configurazioni finali è quindi raggiungibile con una sequenza di mosse lunga al più 3 (dato che ogni numero maggiore di 1 è esprimibile come $2k + 3h$ con h e k interi non negativi).

Quindi, se C è maggiore o uguale a 3, le configurazioni finali cercate sono quelle che, tra le 8 possibili, rispettano le informazioni parziali fornite.

Se invece C è minore di 3, allora le sequenze corrette sono da ricercare in un sottinsieme delle 8 possibili configurazione finali raggiungibili con sequenze di mosse lunghe al più 2 (ovvero le 6 configurazioni finali diverse raggiungibili applicando alla configurazione iniziale una delle 6 sequenze di mosse elencate al punto 2); infine, se C è uguale a 1, la configurazione finale corretta è da ricercare tra le configurazioni raggiungibili a partire da quella iniziale con una sola mossa (le possibili mosse sono elencate al punto 1).