

Sep 04, 12 13:20

elle.txt

Page 1/2

Se, come nel gioco del tetris, consentiamo le rotazioni ma non i ribaltamenti, allora i possibili 4-mini sono precisamente 7 (i 7 pezzi del tetris). Tra questi, i 2 ad L sono uno l'immagine speculare dell'altro:

```

      0           0
la L --> 0       |       0  <-- la L riflessa
          00      |       00

```

Ci sono precisamente 4 tassellazioni distinte del rettangolo di 3 x 8 quadratini per mezzo di 4-mini ad L. Ciascuna di esse impiega ovviamente $3 \times 8 / 4 = 6$ tessere ed e' rappresentata in figura numerando tali tessere da 1 a 6 e riportando su ogni quadratino del rettangolo il numero della tessera che lo ricopre.

```

11344466   12225556   11333466   11333466
12333456   12333456   12344456   12344456
12225556   11344466   12225556   12225556

```

Certo: potessimo ribaltare il rettangolo la tassellazione sarebbe allora unica (la sola rinumerazione delle tessere e' ovviamente ininfluenta), ma nel tetris nessun ribaltamento e' consentito.

Una domanda naturale e' per quali valori di n esistano delle tassellazioni della griglia 3 x n, ma siamo interessati anche al computare t(n), il numero di tassellazioni per ogni dato n.

```

n   0 1 2 3 4 5 6 7 8 9
t(n) 1 0 0 0 0 0 0 0 4 ? ... ? ...

```

Poiche' t(n) cresce rapidamente, e non mi interessa farvi sviluppare la gestione dei grandi numeri, il vostro programma legge dal file "input.txt" un bench di 5 istanze (5 numeri naturali):

```

input.txt
<n1> <n2> <n3> <n4> <n5>

```

e scrive nel file "output.txt" solo le ultime sette cifre decimali del numero di tassellature in una griglia 3 x ni, con i = 1,2,3,4,5. (Ossia scrive i valori t(ni) modulo M, con M=10000000):

```

output.txt
<t(n1)%M> <t(n2)%M> <t(n3)%M> <t(n4)%M> <t(n5)%M>

```

Il motivo per cui ogni singolo caso di prova (ogni singolo input.txt) pone 5 domande diverse per dare punto sta nel fatto che non voglio regalare punti facili ad un programma che risponda sempre 0.

Si assuma che $n1 > n2 > n3 > n4 > n5 \geq 0$ e che n1 possa sempre trovare spazio in un <long long int>.

Ciascuno dei 20 casi di prova (ogni singolo file input.txt) vi fa punto se:

1. $n1 \leq 100000$ e in output.txt azzeccate tutte e 5 le risposte
2. $n1 \geq 100001$ ed in output.txt specificate con degli 0 od 1 se t(ni) e' nullo o positivo.

Ad esempio, sul caso di prova

```

input.txt
100007 8 5 2 0

```

e' la seguente risposta quella che vi consegna il punto

```

output.txt
0 1 0 0 1

```

Sep 04, 12 13:20

elle.txt

Page 2/2

Sappiate, per meglio regolarvi, che in almeno f(n) casi $n1 \leq n$:

```

n   10 20 30 40 50 100 200 1000 10000 100000  max(long long int)
f(n) 1  2  3  4  5  7  8  10  12  16  20

```