



PROBLEMA RE DEI RATTI



*Nous vivrons. Et nous guérirons.
Les cicatrices... Nous les gardons.
Pour ne pas oublier. - Amicia*

Dopo una schiacciante battaglia contro i ratti, è il momento per Amicia e Hugo di fuggire dall'esercito del conte Victor de Arles. I soldati si sono posizionati su un sentiero stretto, che può essere rappresentato come una matrice di dimensioni $2 \cdot n$. Inoltre, sappiamo che su questo sentiero ci sono in totale k soldati.

PROBLEMA Amicia e Hugo definiscono il pericolo di una configurazione come il numero di gruppi di soldati in essa presenti. Più formalmente, consideriamo una matrice binaria $2 \times n$, in cui abbiamo 1 nelle posizioni in cui è presente un soldato. Diciamo che due celle sono connesse se hanno entrambe valore 1 e condividono un lato. Nota che questa relazione è transitiva, nel senso che se due celle a e b sono connesse e le celle b e c sono connesse, allora anche a e c sono considerate connesse. Una componente connessa è un sottoinsieme massimale di celle connesse con valore 1. Il pericolo è il numero di componenti connesse in questa matrice.

Il tuo compito è aiutare i due protagonisti a trovare il valore atteso del pericolo, considerando tutte le possibili configurazioni. Ogni configurazione è considerata equiprobabile. Quindi il valore atteso è uguale alla media del numero di componenti connesse al variare di tutte le possibili configurazioni.

IMPLEMENTAZIONE Devi implementare le seguenti funzioni:

```
void prec(int subtask_id)
```

```
int solve(int n, int k)
```

La prima funzione verrà chiamata una volta all'inizio del grader. Puoi usarla per il preprocessing.

La seconda funzione deve restituire il valore atteso del pericolo, dati i parametri n e k , modulo 998 244 353. Formalmente, sia $M = 998\,244\,353$. Si può dimostrare che la risposta può essere espressa come una frazione irriducibile $\frac{p}{q}$, dove p e q sono interi e $q \not\equiv 0 \pmod{M}$. La funzione deve restituire l'intero uguale a $p \cdot q^{-1} \pmod{M}$. In altre parole, la funzione deve restituire un intero x tale che $0 \leq x < M$ e $x \cdot q \equiv p \pmod{M}$.



La seconda funzione verrà chiamata t volte. Ciò significa che nell'input sono presenti più casi di test!

Non dimenticare di includere il file di intestazione "kor.h", altrimenti otterrai un errore di compilazione!

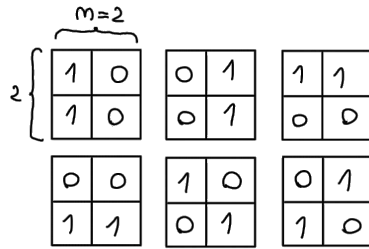
- ASSUNZIONI**
- ◆ $1 \leq t \leq 10$
 - ◆ $1 \leq n \leq 10^9$
 - ◆ $0 \leq k \leq 10^6$
 - ◆ $k \leq 2 \cdot n$

#	Punti	Assunzioni
1	10	$1 \leq n \leq 100$
2	5	$1 \leq n \leq 2000$
3	5	$k \leq 3$
4	15	$k \leq 40$
5	10	$k \leq 400$
6	15	$k \leq 2000$
7	40	Nessuna limitazione aggiuntiva.

ESEMPI

Input	Output	Spiegazione
2	332748119	Nota che al grader devono essere forniti l'indice del subtask, il numero di casi di test t e i valori n, k per ciascun caso di test.
6	1	
2 2	518205646	
5 10	742082393	
2000 3	368118258	
2000 5	937239298	
100 32		
150 278		
7	0	
8	1	
1000000000 0	268791198	
1000000000 1	806373591	
1000000000 2	782159797	
1000000000 3	435727907	
5219873 192	712321002	
853875838 238	257644694	
43782384 1500		
58123292 180000		

Per il primo caso di test del primo esempio, le possibili configurazioni sono mostrate qui sotto:



Ci sono in totale 6 configurazioni, 4 delle quali hanno una sola componente.
 La risposta è quindi $\frac{4 \cdot 1 + 2 \cdot 2}{6} = \frac{4}{3}$.