Problem C: Nice Prefixes

Consider strings formed from characters from an alphabet of size K. For example, if $K = 4$, our alphabet might be *{a,b,c,d}*, and an example string is *bbcac*.

For a string S, define *count(S, k)* to be the number of occurrences of the symbol k in S. For example, *count(bbcac, b) = 2* and *count(bbcac, a) = 1*.

A prefix of a string S is any string obtained from S by deleting some (possibly none) of the trailing characters of S. For example, the prefixes of *acb* are the empty string, *a*, *ac*, and *acb*.

A string S has "nice prefixes" if for every prefix P of S and for every two characters $k1$ and $k2$ in the alphabet, *|count(P, k1) - count(P, k2)| <= 2*. For example, *bbcac* has nice prefixes, but *abbbc* does not because *count(abbb, b) = 3* and *count(abbb, c) = 0*.

Count the number of strings of length L on an alphabet of size K that have nice prefixes. This number can be large, so print its remainder when divided by 1000000007.

Input Specification

The input is a single line containing the two integers L and K, separated by spaces, with $1 <= L <= 10^{18}$ and $1 <= K <= 50$.

Sample Input

4 2

Output Specification

Output a single line containing the number of strings of length L on an alphabet of size K that have nice prefixes, modulo 1000000007.

Output for Sample Input

12

Malcolm Sharpe, Ondřej Lhoták