

## F. Egg Roulette

time limit per test: 3 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

The game of Egg Roulette is played between two players. Initially  $2R$  raw eggs and  $2C$  cooked eggs are placed randomly into a carton. The shells are left on so there is no way to distinguish a raw egg from a cooked egg. One at a time, a player will select an egg, and then smash the egg on his/her forehead. If the egg was cooked, not much happens, but if the egg was raw, it will make quite the mess. This continues until one player has broken  $R$  raw eggs, at which point that player is declared the loser and the other player wins.

The order in which players take turns can be described as a string of 'A' and 'B' characters, where the  $i$ -th character tells which player should choose the  $i$ -th egg. Traditionally, players take turns going one after the other. That is, they follow the ordering "ABABAB . . .". This isn't very fair though, because the second player will win more often than the first. We'd like you to find a better ordering for the players to take their turns. Let's define the unfairness of an ordering as the absolute difference between the first player's win probability and the second player's win probability. We're interested in orderings that minimize the unfairness. We only consider an ordering valid if it contains the same number of 'A's as 'B's.

You will also be given a string  $S$  of length  $2(R + C)$  containing only 'A', 'B', and '?' characters. An ordering is said to match  $S$  if it only differs from  $S$  in positions where  $S$  contains a '?'. Of the valid orderings that minimize unfairness, how many match  $S$ ?

### Input

The first line of input will contain integers  $R$  and  $C$  ( $1 \leq R, C \leq 20, R + C \leq 30$ ).

The second line of input will contain the string  $S$  of length  $2(R + C)$  consisting only of characters 'A', 'B', '?'.

### Output

Print the number of valid orderings that minimize unfairness and match  $S$ .

### Examples

<b>input</b>
1 1 ??BB
<b>output</b>
0

  

<b>input</b>
2 4 ?BA??B??A???
<b>output</b>
1

  

<b>input</b>
4 14 ????A??BB????????????AB????????????
<b>output</b>
314

### Note

In the first test case, the minimum unfairness is 0, and the orderings that achieve it are "ABBA" and "BAAB", neither of which match  $S$ . Note that an ordering such as "ABBB" would also have an unfairness of 0, but is invalid because it does not contain the same number of 'A's as 'B's.

In the second example, the only matching ordering is "BBAAABABABBA".