


# Linguagem de Blogger

Por Pablo Heiber, Fidel Schaposnik e Rafael Garcia  Argentina

Timelimit: 8

Brenda, neta de Benjamin, tem um blog onde ela posta artigos sobre a escola, os amigos e outras questões da vida. Intrigado com suas opiniões, Benjamin tentou lê-lo, mas logo se deu conta de que era muito difícil de ler por causa das peculiaridades da escrita de Brenda.

Brenda escreve sem espaços ou sinais de pontuação, e, além disso, ela usa letras minúsculas e maiúsculas de uma forma liberal e estranha. Por exemplo, uma das suas mensagens é "PrOgRAMmINgISgrEAt". Benjamin tem dificuldade em perceber as palavras "programming", "is" e "great" quando são escritos desta forma.

Para melhorar a sua compreensão, Benjamin decidiu fazer o seguinte: ele vai primeiro escolher uma determinada string  $T$  e um post do blog no qual ele está interessado e, em seguida, ele irá selecionar uma substring contígua do post e irá procurar  $T$  dentro da substring, de uma forma case-insensitive. Para cada ocorrência de  $T$  dentro da substring, ele vai calcular o número de incompatibilidade de case e, finalmente, ele vai obter o máximo entre todos esses valores. Por exemplo, se Benjamin escolhe "GR", como  $T$  e, em seguida, seleciona a substring "PrOgRAM", ele iria encontrar uma ocorrência única "gR" para o qual o número de incompatibilidade de case é 1. Pela mesma subsequência, se "r" foi escolhido como  $T$ , ele teria encontrado duas ocorrências, "r" com 0 incompatibilidades e "R", com uma incompatibilidade, portanto, o número máximo de incompatibilidades seria 1.

Para complicar mais as coisas, Brenda incluiu no blog um script que, depois da operação com a seleção de uma substring, inverte o "case" de todas as letras selecionadas. Isto significa que após selecionar "PrOgRAM" e proceder como explicado acima, o post exemplo seria lido como "pRoGrammINgISgrEAt". Se Benjamin selecionar "ammiNgi" como uma segunda substring, após calcular o seu resultado o post será deixado como "pRoGrAMMinGISgrEAt", acumulando as duas inversões.

Você receberá a string  $T$  e o texto original do post do blog escolhido por Benjamin. A você serão também fornecido uma lista com as seleções de substring feitas por Benjamin, a fim de fazê-las. Você precisa calcular, para cada seleção, o número máximo de cases incompatíveis das ocorrências de  $T$  na parte selecionada, considerando todas as inversões feitas pelas seleções anteriores. Observe que a inversão de case ocorre após o cálculo do resultado de cada seleção.

## Entrada

A entrada contém vários casos de teste e termina com EOF. A primeira linha de um caso de teste contém um  $N$  ( $1 \leq N \leq 10^5$ ) e uma string não vazia  $T$  com no máximo 5 letras, representando respectivamente o número de seleções de substring e a string a ser selecionada. A segunda linha contém uma string não vazia  $P$  com no máximo  $10^5$  letras, indicando o texto original postado no blog. Posições do post são numerados com inteiros consecutivos da esquerda para a direita, sendo 1 a posição mais à esquerda e  $IP$  a posição mais à direita. Cada uma das próximas  $N$  linhas descreve uma seleção de substring com 2 inteiros  $L$  e  $R$  ( $1 \leq L \leq R \leq IP$ ) indicando que a substring inicia na posição  $L$  e termina na posição  $R$ , inclusive.

## Saída

Para cada caso de teste imprima  $N$  linhas, cada uma delas contendo um inteiro. Na  $i$ -ésima ( $i$ -th) linha escreva o número máximo de casos de compatibilidade das ocorrências de  $T$  nas  $i$ -ésimas ( $i$ -th) seleções de substring, considerando todas as inversões de case feitas pelas seleções anteriores. Se tal ocorrência não existir, imprima o valor  $-1$ .

---

Exemplo de Entrada

---

Exemplo de Saída

---

3	gR	0
	PrOgRAMmINgiSgrEAt	2
1	7	-1
4	18	2
6	14	4
9	abCAb	1
	aBcAbCAbaBCAb	-1
1	13	0
1	13	5
4	8	2
5	11	-1
3	11	4
4	10	
1	13	
8	8	
1	13	