



Jak najwięcej par



Tym razem Brajanek biedzi się nad długim ciągiem znaków: musi znaleźć, ile razy w nim występuje zadany dwuznakowy *podciąg*, czyli para znaków. Na przykład dla ciągu **abcdad** i pary **ad** widzimy trzy takie wystąpienia:

abcdad **abcdad** **abcdad**.

Jak widać, podciąg musi być utworzony przez znaki oryginalnego ciągu, bez zmiany kolejności.

Brajanek ma jednak pewne utrudnienie: mianowicie ma podaną liczbę naturalną k , która określa, ile (maksymalnie) znaków oryginalnego ciągu można zmienić na dowolne inne. To znacząco poszerza możliwości dopasowania podciągu, no ale zwiększa też niepomniernie stopień komplikacji zadania. Dla powyższego przykładu, jeśli $k = 2$, wtedy możemy dokonać następującej podmiany:

abcdad \rightarrow **aaddad**.

Teraz możemy doszukać się aż siedmiu wystąpień podciągu **ad**:

aaddad **aaddad** **aaddad**
aaddad **aaddad** **aaddad** **aaddad**.

Napisz odpowiedni program, który pomoże Brajankowi w rozwiązaniu jego zadania.

Dane wejściowe

Pierwszy wiersz danych wejściowych zawiera liczby naturalne n, k ($2 \leq n \leq 200$, $0 \leq k \leq n$) oznaczające odpowiednio długość ciągu oraz liczbę możliwych zamian znaków.

W drugim wierszu zapisane jest n -znakowy ciąg złożony z małych liter alfabetu łacińskiego.

W trzecim wierszu zapisany jest dwuznakowy ciąg złożony z małych liter alfabetu łacińskiego – to jest właśnie ten podciąg, którego będziemy poszukiwać.

Wynik programu

Program powinien wypisać wiersz tekstu zawierający liczbę możliwych wystąpień podciągu po wykonaniu maksymalnie k podmian znaków w oryginalnym ciągu.

Przykład

Dla danych wejściowych:

```
6 2  
abcdad
```

prawidłowym wynikiem jest:

```
7
```

Jest to przykład z treści zadania.

Dla danych wejściowych:

```
7 2  
xzxyzzx  
xx
```

prawidłowym wynikiem jest:

```
15
```

W tym przykładzie należy dążyć do otrzymania ciągu **xxxxyxxx**.