# How Many Substrings?

Consider a string of $n$ characters, $s$, of where each character is indexed from $0$ to $n-1$.

You are given $q$ queries in the form of two integer indices: $left$ and $right$. For each query, count and print the number of different substrings of $s$ in the inclusive range between $left$ and $right$.

**Note:** Two substrings are *different* if their sequence of characters differs by at least one. For example, given the string $s =$ `aab`, substrings $s_{[0,0]} =$ `a` and $s_{[1,1]} =$ `a` are the same but substrings $s_{[0,1]} =$ `aa` and $s_{[1,2]} =$ `ab` are different.

### Input Format

The first line contains two space-separated integers describing the respective values of $n$ and $q$.
The second line contains a single string denoting $s$.
Each of the $q$ subsequent lines contains two space-separated integers describing the respective values of $left$ and $right$ for a query.

### Constraints

- $0 \leq left \leq right \leq n-1$
- String $s$ consists of lowercase English alphabetic letters (i.e., `a` to `z`) only.

### Subtasks

- For $30\%$ of the test cases, $1 \leq n, q \leq 100$
- For $50\%$ of the test cases, $1 \leq n, q \leq 3000$
- For $100\%$ of the test cases, $1 \leq n, q \leq 10^5$

### Output Format

For each query, print the number of different substrings in the inclusive range between index $left$ and index $right$ on a new line.

### Sample Input 0

```
5 5
aabaa
1 1
1 4
1 1
1 4
0 2
```

### Sample Output 0

```
1
8
1
8
5
```

### Explanation 0

Given $s =$ `aabaa`, we perform the following $q = 5$ queries:

1. `1 1`: The only substring of `a` is itself, so we print `1` on a new line.
2. `1 4`: The substrings of `abaa` are `a`, `b`, `ab`, `ba`, `aa`, `aba`, `baa`, and `abaa`, so we print `8` on a new line.
3. `1 1`: The only substring of `a` is itself, so we print `1` on a new line.
4. `1 4`: The substrings of `abaa` are `a`, `b`, `ab`, `ba`, `aa`, `aba`, `baa`, and `abaa`, so we print `8` on a new line.
5. `0 2`: The substrings of `aab` are `a`, `b`, `aa`, `ab`, and `aab`, so we print `5` on a new line.