CS201c Programming Evaluation 3
Instructor: Apurva Mudgal
Topic: String matching using hashing
Due Date: Nov 3, 2019 by 11:59 pm

Problem description.

Input Format:

The first line is a single positive integer n.

The second line contains a bit string x of length exactly n.

The third line contains a second positive integer m.

This is followed by m lines, with p-th (1<=p<=m) line containing four positive integers i_p, j_p, k_p, and l_p between 1 and n:

i_1 j_1 k_1 l_1
i_2 j_2 k_2 l_2
.
.
.
i_m j_m k_m l_m

Output Format:

The output consists of m lines.

The p-th line has the single integer 1 if and only if substring x[i_p … j_p] is equal to substring x[k_p … l_p]. Otherwise, the p-th line has the single integer 0.

Note. You can use a randomized algorithm i.e., your C++ program can have a random supply of random bits using srand() and rand() functions. *Be sure to initialize your random number generator with a fresh seed using srand() at the start of your C++ program.*

Hint. Find an efficient randomized, hashing scheme for substrings such that two unequal substrings can have the same hash value only with some maximum probability.

Requirements:

1. *Running time.* Worst-case time taken by your C++ program should be O((n+m) \log_2(n)) in the RAM model.

2. *Probability of giving a correct answer.* Further, for each 1 <= p <= m, your C++ should satisfy the following condition:

   Pr [the p-th line of your output is correct] >= 1-(1/n)

3. You cannot use any in-built libraries (including standard template library). All data structures should be implemented in C++ from scratch.

4. **Collaboration is not permitted on this assignment. Your submitted code should be completely your own.**

   See section titled ``Honor Code" in course outline already shared with you.

Example.

Sample Input.

```
10
1011011011
6
1 4 7 10
2 5 7 10
2 4 5 7
2 4 8 10
3 7 6 10
1 5 6 10
```

Correct Sample Output.

```
1
0
1
1
1
0
```

Note. Since your C++ code is randomized, you are allowed to make an error on each output with probability at most ⅙=0.167.