# Question 1:

**What is the worst case time complexity of Rabin Karp algorithm, where N is the size of the text and M is the size of the pattern?**

#### Select the correct choice:

O(M + N). In every case, the algorithm is linear

O(MxN). Inputs that generate same hash values make this close to brute-force

O(M ^ N). Poor hash function makes this an exponential algorithm

O(N - M). The algorithm is very efficient and linear

# Answer 1:

O(MxN). Inputs that generate same hash values make this close to brute-force.

# Question 2:

**Which of the following problems can potentially use Rolling Hash as a solution?**

#### Select the correct choices:

Given two strings A and B, and a number X find if they have a common sequence of length X.

Given two strings, find out if one is a rotation of the other.

Swap two integers without using a temporary variable.

Print all values in a BST in sorted order

Keep two files on disk in sync, by copying only the deltas.

# Answer 2:

Given two string A and B, and a number X find if they have a common sequence of length X.

Given two strings, find out if one is a rotation of the other

Keep two files on disk in sync, by copying only the deltas.

# Question 3:

**What technique does Rolling Hash use, in order to avoid overflow and still minimize conflicts?**

#### Select the correct choice:

Modulo Prime each numerical operation

Divide each number by INT\_MAX before the operation

Subtract INT\_MAX from each number before the operation

Multiply each number by a fraction between 0 and 1

# Answer 3:

Modulo Prime each numerical operation