

Q1) Write a program to check if a given number is a Rotational prime. A Rotational prime is a prime number that remains prime under all rotations of its digits.

Example 1: 1193

Rotations: 1193, 1931, 9311, 3119

All rotations are prime.

Result: 1193 is a Rotational prime.

Example 2: 197

Rotations: 197, 971, 719

All rotations are prime.

Result: 197 is a Rotational prime.

Example 3: 37

Rotations: 37, 73

Both rotations are prime.

Result: 37 is a Rotational prime.

Example 4: 23

Rotations: 23, 32

32 is not a prime number.

Result: 23 is not a Rotational prime.

Example 5: 31

Rotations: 31, 13

Both rotations are prime.

Result: 31 is a Rotational prime.

Example 6: 11

Rotations: 11, 11

Both rotations are prime.

Result: 11 is a Rotational prime.

Example 7: 79

Rotations: 79, 97

Both rotations are prime.

Result: 79 is a Rotational prime.

Example 8: 17

Rotations: 17, 71

Both rotations are prime.

Result: 17 is a Rotational prime.

Example 9: 101

Rotations: 101, 011 (11), 110

110 is not a prime number.

Result: 101 is not a Rotational prime.

Example 10: 73

Rotations: 73, 37

Both rotations are prime.

Result: 73 is a Rotational prime.

Q2) Write a program that generates all permutations of a given string using loops (avoid using recursion or built-in functions).

Example Output:

For the string "ABC":

Permutations of 'ABC':

ABC

BAC

BCA

ACB

CAB

CBA

Example 1:

Input: "AB"

Output:

Copy code

AB

BA

Example 2:

Input: "123"

Output:

Copy code

123

213

231

132

312

321

Example 3:

Input: "XYZ"

Output:

Copy code

XYZ

YXZ

YZX

XZY

ZXY

ZYX