* WRITE A PROGRAM TO CHECK WHETHER STRINGS ARE ROTATIONS OF EACH OTHER.

Input: input\_string and rotated\_string are two string variables

Output: if rotated\_string is a rotation of input\_string then return true else return false.

Algorithm: ---\* Method – I \*---

**Step - I:** if length of input\_string is not equal to rotated\_string then return false

**Step – II:** if length of input\_string is equal to rotated\_string then create a temporary string and store the concatenation value (input\_string + input\_string) to temporary string

**Step – III:** if rotated\_string is a substring of temporary string then input\_string and rotated\_string are rotations of each other therefore return true else return false.

Algorithm: ---\* Method – II \*---

**Step - I:** if length of input\_string is not equal to rotated\_string then return false

**Step – II:** if length of input\_string is equal to rotated\_string then loop through each character in rotated\_string and find the first character of input\_string in rotated\_string

* if found then match the corresponding characters of rotated\_string in cyclic order with input\_string and if every character is present in same order then return true else end the current iteration and start next iteration.
* if not found then end the current iteration and start with next iteration.

**Step – III:** return false stating that rotated\_string is not a rotation of input\_string.