**Cyclic sort**

**\*\*1. This Problem can best solved with cyclic sort because all indices lies between 0 to n-1 and the swaps also follows the same pattern.**

**2. This will help you achieve O(1) Space complexity and with minimum no of swaps you can achieve your target**

**\*\***

class Solution {

public:

string restoreString(string s, vector<int>& indices) {

for(int i=0;i<indices.size();i++)

{

while(indices[i]!=i)

{

swap(s[i],s[indices[i]]);

swap(indices[i],indices[indices[i]]);

}

}

return s;

}

};