Lecture-21

Q1. 189. Rotate Array

```
class Solution {
public:
void rotate(vector<int>& nums, int k) {
vector<int> temp(nums.size());
for(int i=0;i<nums.size();i++){
temp[(i+k)%nums.size()] = nums[i];
}
nums = temp;
}
};
```

Q2. 1752. Check if Array Is Sorted and Rotated

```
class Solution {
public:
bool check(vector<int>& nums) {
int count=0;
for(int i=1;i<=nums.size();i++){
if(nums[(i-1)%nums.size()]>nums[i%nums.size()]){
count++;
}
}
if(count==1 || count==0){
return true;
}
return false;
}
};
```

Q3. **Sum Of Two Arrays**
https://www.codingninjas.com/studio/problems/sum-of-two-arrays_893186

```
#include <bits/stdc++.h>
vector<int> reverse(vector<int> v){
int s=0;
int e = v.size()-1;
while(s<=e){
swap(v[s],v[e]);
```

```
e--;
}
return v;
}
vector<int> findArraySum(vector<int>&a, int n, vector<int>&b, int m) {
// Write your code here.
int i=n-1;
int j = m-1;
int carry=0;
vector<int> ans;
while(i>=0 && j>=0){
int val1 = a[i];
int val2 = b[j];
int sum = val1+val2+carry;
carry = sum/10;
sum = sum%10;
ans.push_back(sum);
i--;
j--;
}
while(i>=0){
int sum = a[i]+carry;
carry = sum/10;
sum = sum%10;
ans.push_back(sum);
i--;
}
while(j>=0){
int sum = b[j]+carry;
carry = sum/10;
sum = sum%10;
ans.push_back(sum);
j--;
}
while(carry!=0){
int sum = carry;
carry = sum/10;
sum = sum%10;
ans.push_back(sum);
}
return reverse(ans);
}
```