**CP Problem Statement : 1**

**A Job Ready Bootcamp in C++, DSA and IOT**

**MySirG**

1. **Pallindrome Integer**

Problem Statement

Given an integer x , return true if x is palindrome integer.

An integer is a palindrome when it reads the same backward as forward.

For example, 121 is a palindrome while 123 is not.

Example 1:

Input: x = 121

Output: true

Explanation: 121 reads as 121 from left to right and from right to left.

Example 2:

Input: x = -121

Output: false

CP Problem Statement : 1 2

Explanation: From left to right, it reads -121. From right to left, it becomes 121-. There

fore it is not a palindrome.

Example 3:

Input: x = 10

Output: false

Explanation: Reads 01 from right to left. Therefore it is not a palindrome.

Constraints:

-231 <= x <= 231 - 1

Note : Solve it without converting the integer to a string.

**Sol –**

#include<iostream>

using namespace std;

int palindrome(int);

int main()

{

int x;

cout<<"Enter a number : ";

cin>>x;

if(x<0)

cout<<"False";

else if(palindrome(x))

cout<<"True";

else

cout<<"False";

return 0;

}

int palindrome(int n)

{

int rev,rem=0,temp=n;

while(temp!=0)

{

rem=temp%10;

rev=rev\*10+rem;

temp/=10;

}

if(rev==n)

return 1;

return 0;

}