IMP Q/S

Q1 Palindrome.

Ans Use code that used to reverse a number.

Q2 GCD or LCM [GCD → gretest number divided by both numbers]

Ans gcd fn already has a fn in namespace std.

```
int gcd(long long x, long long y){
    if(y==0){
        return x;}
    else{
        gcd(y, x%y);
```

for LCM.

lcm = (A*B)/gcd(A,B)

79

0.1
100) 120
100
20

112
10x10

Date..... Day.....

Q3 Armstrong Number.

Ans ①
```
int numberofdigits(int x)
{
    int n=0;
    while(x>0){
        n++;
        x=x/10;
    }
    return n;
}
```

Dry Run

① x=153 x=15 x=1 x=0
n=1 n=2 n=3

② temp=153
r=3
pow(3,3)=27
S=27

→ temp=15

Date..... Day.....

②
```
bool isArmstrong(int x){
    int n= number of digit(x);
    int temp=x, sum=0;
    while(temp>0).
    { int r=temp%10;
      sum= Sum+ pow(r,n);
      temp= temp/10;
    }
    return (sum == x);
```

pow(5,3)
S=125+27=152

→ temp=1
r=1
pow(1,3)
S=153

Parent's Sign. .....

Teacher Sign. .....

Date..... Day.....

Q4 Factorial of N

Ans
```
int fact(int n)
if(n==0){
    return 1;
}
else{
    return n*fact(n-1);}
```

Q5 Reverse an Array.

Ans Approach: move from n → 1

Date..... Day.....

```
for(int i=n-1;i>=0;i--){
    cout<<arr[i]<<" ";}
```

Q6 Palindrome of strings.

Ans
```
int isPalindrome(string S)
{
    int j=S.length()-1
    for(int i=0;i<S.length()/2;i++,j--){
        if(S[i]!=S[j]){
            return 0;
        }
    }
    return 1;
}
```

Teacher Sign. ..... Parent's Sign. .....

Date..... Day.....

Q7 Palindrome number.

Ans

```
string is_Palindrome(int n)
{
    int rev=0; ans=0
    int temp=n;
    while(n!=0){
        rev = n%10
        ans = ans*10 + rev;
        n /=10;
    }
    if(temp==ans){
        return "Yes";
    }
    else{
        return "No";
    }
```

Date..... Day.....

Q8 Fibonacci number.

Ans

```
int fib(int n){
    if(n<=1){
        return n;
    }
    else{
        return fib(n-1)+fib(n-2);
    }
}
```

Parent's Sign. ..... Teacher Sign. .....