Name: SUNDEEP A	SRN: PES1UG20CS445	Section: O
	Date: 20-5-21	Week Number: 3

1	**Write a function to reverse a given number and check whether a given number is palindrome or not.** **Input:** Enter the number 121 **Output:** The Number 121 is Palindrome **Input:** Enter the number **Output:** 123 Number 123 is Not Palindrome
	Program:

```c
#include<stdio.h>
int reverse(int);
int main()
{
        int n,r;
        printf("Enter the number\n");
        scanf("%d",&n);
        r=reverse(n);
        if(n==r)
                printf("The Number %d is a Palindrome\n",n);
        else
                printf("The number %d is not a Palindrome\n",n);
        return 0;
}
int reverse(int n)
{
        int digit,r;
        r=0;
        while(n!=0)
        {
                digit = n%10;
                n=n/10;
                r=r*10+digit;
        }
        return r;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 1.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc 1.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter the number
121
The Number 121 is a Palindrome

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter the number
123
The number 123 is not a Palindrome
```

2 **Write a C program to compute GCD of three numbers using functions.**

Input:

Enter the values of a,b and c

10 4 16

Output:

GCD(10,4,16)=2

Program:

```
#include<stdio.h>
int gcd(int ,int);
int main()
{
        int a,b,c,d,res;
        printf("Enter the values of a,b and c\n");
        scanf("%d%d%d",&a,&b,&c);
        d=gcd(a,b);
        res=gcd(d,c);
        printf("GCD of (%d,%d,%d) = %d",a,b,c,res);
}
int gcd(int m ,int n)
{
        while(m!=n)
        {
                if(m>n)
                        m=m-n;
                else
                        n=n-m;
        }
        return m;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 2.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc 2.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter the values of a,b and c
10 4 16
GCD of (10,4,16) = 2
```

3 Write a program in C to check Armstrong and perfect numbers using functions.

Input:

Input any number: 153

Output:

The 153 is an Armstrong number.

The 153 is not a Perfect number.

Input:

Input any number: 28

Output:

The 28 is not an Armstrong number.

The 28 is a Perfect number.

Program:

```
#include<stdio.h>
int checkArmstrong(int);
int checkPerfect(int);
int main()
{
        int n1;
        printf("Input any number\n");
        scanf("%d",&n1);
        if(checkArmstrong(n1))
                printf("The given number %d is Armstrong number \n",n1);
        else
                printf("The given number %d is not a Armstrong number \n",n1);
        if(checkPerfect(n1))
                printf("The given number %d is Perfect number \n",n1);
        else
                printf("The given number %d is not a Perfect number \n",n1);
        return 0;
}
int checkArmstrong(int n1)
{
        int d,sum=0,num=n1;
        while(num!=0)
        {
                d=num % 10;
                sum+=d*d*d;
                num=num/10;
        }
        return(n1==sum);
}
```

```
int checkPerfect(int n1)
{
        int i,sum=0,num=n1;
        for(i=1;i<num;i++)
                if(num%i ==0)
                        sum+=i;
        return (sum == n1);
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 3.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc 3.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Input any number
153
The given number 153 is Armstrong number
The given number 153 is not a Perfect number

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Input any number
28
The given number 28 is not a Armstrong number
The given number 28 is Perfect number
```

4 Write a program in C to check whether a number is a prime number or not using function

Input:

Input a positive number : 12

Output:

The number 12 is not a prime number

Input:

Input a positive number : 13

Output:

The number 13 is a prime number

Program:

```
#include<stdio.h>
int PrimeorNot(int);
int main()
{
        int n1;
        printf("Enter the number");
        scanf("%d",&n1);
        if(PrimeorNot(n1)==1)
                printf("The number %d is a prime number \n",n1);
        else
                printf("The given number %d is not a prime number \n",n1);
        return 0;
}
int PrimeorNot(int n1)
{
        int i=2;
        while(i<=n1/2)
                if(n1%i==0)
                        return 0;
                else
                        i++;
        return 1;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 4.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc  4.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter the number 12
The given number 12 is not a prime number

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter the number 13
The number 13 is a prime number
```

5 Write a program in C to convert decimal number to octal number using function

Input:

Input any decimal number : 25

Output:

Equivalent Octal Number: 31

Input:

Input any decimal number : 15

Output:

Equivalent Octal Number: 17

Program:

```
#include<stdio.h>
#include<math.h>
int deciamloOctal(int);
int main()
{
        int decimalnum;
        printf("Enter a decimal number : \n");
        scanf("%d",&decimalnum);
        printf("Equivalent Octal Number : %d",deciamloOctal(decimalnum));
        return 0;
}
int deciamloOctal(int decimalnum)
{
        int octalnum=0,temp=1;
        while(decimalnum !=0)
        {
                octalnum = octalnum + (decimalnum %8) * temp;
                decimalnum=decimalnum/8;
                temp = temp *10;
        }
        return octalnum;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 5.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc  5.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
Enter a decimal number :
25
Equivalent Octal Number : 31
```

6 Write a program in C to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 using function.

Output:

The sum of the series is : 34

Program:

```
#include<stdio.h>
int fact(int);
int main()
{
        int sum;
        sum = fact(1)/1 + fact(2)/2 +fact(3)/3 + fact(4)/4 +fact(5)/5;
        printf("The sum of the series is : %d\n",sum);
        return 0;
}
int fact(int n)
{
        int num=0,f=1;
        while(num<=n-1)
        {
                f=f+f*num;
                num++;
        }
        return f;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc -c 6.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>gcc  6.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 3>a
The sum of the series is : 34
```