SSN COLLEGE OF ENGINEERING (Autonomous)

Affiliated to Anna University

DEPARTMENT OF CSE

UCS 1211 PROGRAMMING IN C LABORATORY

Assignment 2

Modular Programming with C

Reg Number : 185001131

Name : Sai Charan B

Class : CSE - B

1. Modify A1(1) to have a function CheckOddEven(num) that checks if the num is odd or even; sets a flag accordingly and return it. Use this function to find the sum of even and odd numbers in a given input of N numbers.

```
#include <stdio.h>

void main()
{       int num,i,n,osum,esum,flag;
        flag=0;osum=0;esum=0;
        printf("Enter the number of numbers you want to enter ?");
        scanf("%d",&n);
        for(i=0;i<n;i++)
        {       scanf("%d",&num);
                flag=CheckOddEven(num);
                if(flag==1)
                        esum+=num;
                else if(flag==0)
                        osum+=num;
```

```
        }
        printf("Oddsum: %d",osum);
        printf("Evensum: %d",esum);
}

int CheckOddEven(int num)
{       int flag=0;
        if(num%2==0)
                flag=1;
        return(flag);
}
```

Output:

```
cseb131@jtl-29:~$ ./oddeve
Enter the number of numbers you want to enter ?3
1
2
6
Oddsum: 1Evensum: 8
```

2. Write a C function ReverseNum(num) that takes integer num and reverses its digits. Let num be passed by reference.

```
#include <stdio.h>
#include <math.h>

void main()
{       int n;
        printf("Enter a number: ");
        scanf("%d",&n);
        printf("Reversed number = %d ",ReverseNum(n));
}

int ReverseNum(int n)
{       int temp=n,j=0,i=0,sum=0,x;
        while(temp>0)
         {      j++;
```

```
                temp/=10;
        }
        while(n)
        {       i++;
                x=n%10;
                sum+=(x*pow(10,j-i));
                n/=10;
        }
return(sum);
}
```

Output:

cseb131@jtl-29:~$ ./rev

Enter a number: 257825

Reversed number = 528752

3. Write a function power(X,N) that will allow a floating-point number to be raised to an integer power. Y = X N

```
#include <stdio.h>

void main()
{       int n;
        float x;
        printf("Enter the number: ");
        scanf("%f",&x);
        printf("Enter the exponent: ");
        scanf("%d",&n);
        power(x,n);
}

void power(float x,int n)
{       int i,flag=0;
        float prod=1.000;
        if(n<0)
        {       n*=-1;
                flag=1;
        }
        for(i=1;i<=n;i++)
                prod*=x;
        if(flag==1)
                prod=1/prod;
        printf(" %f power %d is: %f ",x,n,prod);

}
```

Output:

```
cseb131@jtl-29:~$ ./pow

Enter the number: 5.6

Enter the exponent: 2

5.600000 power 2 is: 31.359999
```

4. Find the product of n floating point numbers. The numbers should be read from the keyboard.

```
#include<stdio.h>
float product(float x,float n)
{       printf("enter number. 0 to stop  ");
        scanf("%f",&n);
        if (n==0)
                return x;
        else
                x=x*n;
        return product(x,n);
}
void main()
{       float x=1,n=1,z;
        z=product(x,n);
        printf(" \nThe ans is %f",z);
}
```

Output:

```
cseb131@jtl-29:~$ ./float
enter number. 0 to stop  6.4
enter number. 0 to stop  8.2
enter number. 0 to stop  4.77
enter number. 0 to stop  0

The product is 250.329590
```

5. Write a recursive function that reads N and prints from N to 0.

```
#include<stdio.h>

void main()
{       int n;
        printf("Enter number: ");
        scanf("%d",&n);
        print(n);
}


void print(int n)
{       printf("%d",n);
        if(n>=1)
                printdigit(n-1);
}
```

Output:

```
cseb131@jtl-29:~$ ./rec
Enter number: 6
6543210
```

6.The factorial of an integer n, written n!, is the product of all the integers from 1 to n inclusive. The factorial quickly becomes very large; 13! is too large to store as an integer on most computers, and 35! is too large for a floating-point variable. Your task is to find the rightmost non-zero digit of n!.

```
#include<stdio.h>
int factright(int n)
{       int fact=1,i,x;
        for (i=1;i<=n;i++)
        {       fact*=i;
        }
        x=fact%10;
        if (x!=0)
        {       return x;
        }
        else
        {       while(x==0)
        {       fact=fact/10;
                x=fact%10;
        }
                return x;
        }
}
void main()
{       int n,r;
        printf("Enter");
        scanf("%d",&n);
        r=rightfact(n);
        printf("%d",r);
}
```

Outcome :

```
cseb131@jtl-29:~$ ./factorial
Enter5
4
```