

UCS 2201 Fundamentals and Practice of Software Development

A2: Practicing Looping Constructs of C

Batch 2023-2027

Name: SK Shivaanee

Register ID: 2310257

Section: CSE B

Dr. Chitra Babu, Dr. D. Thenmozhi, Dr. M. Saritha

Learning Outcome: ● You will be able to apply computational thinking to solve problems that can be solved by processing its own smaller instances using recursive functions.

Best Practices: Naming conventions to be followed. Use comment statements. Avoid reading and printing statements within function definition

Assignment:

Solve the following problems by implementing in C using recursive functions. (CO7, K3, 1.3.1, 1.4.1, 2.1.2, 2.1.3, 4.2.1, 14.2.1, 14.2.2).

- Identify the possible input and return value for each of the functions.
- Write pseudo code for all the functions.
- Develop a program to test the functions with atleast two test cases.

1.	20/02/2024	Basic programming constructs of C.	24/2/24
2.	02.03.2024	Practicing looping constructs of C.	5/3/24
3.	14.03.2024	Programs using user-Defined functions.	19/3/24
4.	19.03.2024	Programs using recursive functions.	19/3/24

1. Use a recursive function to generate the first n fibonacci numbers. Write a function to find OddFibSum and EvenFibSum from the first n fibonacci numbers. OddFibSum will have the sum of odd numbers in the Fibonacci series and EvenFibSum will have the sum of even numbers in the Fibonacci series.

Pseudocode:

→ For function fibo(int n)

1. Begin
2. if (n==0 or n==1)
 return n;
3. else
 return (fibo(n-1)+fibo(n-2));
4. End

→ For function findSum (int n, int *esum, int *osum)

1. Begin
2. if (n%2 == 0)
 *esum = *esum + n;
3. else
 *osum = *osum + n;
4. End.

```
#include<stdio.h>
int fibo(int n)
{
    if(n==0 || n==1)
        return n;
    else
        return (fibo(n-1)+fibo(n-2));
}
void findSum(int n,int *esum, int*osum)
{
    if(n%2==0)
        *esum=*esum+n;
    else
        *osum=*osum+n;
}
int main()
{
    int n, esum=0,osum=0,p;
    printf("Enter n \n");
    scanf("%d",&n);
    for(int i=0;i<=n;i++)
    {
        p=fibo(i);
        printf("%d ",p);
        findSum(p,&esum,&osum);
    }
    printf("\n Even sum : %d",esum);
    printf("\n Odd sum : %d",osum);
    return 0;
}
```

```

Enter n
5
0 1 1 2 3 5
Even sum : 2
Odd sum : 10

```

```

Enter n
7
0 1 1 2 3 5 8 13
Even sum : 10
Odd sum : 23

```

```

Enter n
10
0 1 1 2 3 5 8 13 21 34 55
Even sum : 44
Odd sum : 99

```

2. Write a recursive function to find power(n,p) which can be computed as $n * \text{power}(n, p-1)$. Write a function to compute the following series using the power function for the given n and p.

$$1^p + 2^p + 3^p + \dots + n^p$$

pseudocode:

```

→ for function power(int n, int p)
1. Begin
2. if (p == 0)
    return n;
3. else
    return (n * power(n, p-1));
4. End.

→ for function series(int n1, int p1)
1. Begin
2. if (n1 == 1)
    return 1;
3. else
    return (power(n1, p1) + series(n1-1, p1));
4. End.

```

```

#include<stdio.h>
#include<math.h>
int power(int n, int p)
{
    if(p==1)
        return n;
    else
        return (n*power(n,p-1));
}
int series(int n1, int p1)
{
    int sum=0;
    if(n1==1)
        return 1;
    else
        return (power(n1,p1)+series(n1-1,p1));
}
int main()
{
    int n,p;
    printf("Enter n and p");
    scanf("%d %d",&n,&p);
    int result=series(n,p);
    printf("%d",result);
    return 0;
}

```

Enter n and p3	Enter n and p5	Enter n and p5
4	7	1
98	96825	15

3. Write a recursive function to find fact(n). Write a function to compute the Cos series using fact function and power function defined in Qn. 2.

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!}$$

Pseudo code:

→ For the function fact(int n)

1. Begin
2. if (n == 1)
 return 1;
3. else
 return (n * fact(n-1));
4. End.

→ For the function power(int n, int p)

1. Begin
2. if (p == 1)
 return n;
3. else
 return (n * power(n, p-1));
4. End.

→ For the function cos(int x, int n)

1. Begin
2. float sum ← 0;
3. for i = 1 to n
 sum = sum + (power(-1, i) * power(x, 2*i) / (fact(2*i)));
 end for
4. return sum;
5. End.

```

#include<stdio.h>
int fact(int n)
{
    if (n==1)
        return 1;
    else
        return (n*fact(n-1));
}
int power(int n, int p)
{
    if(p==1)
        return n;
    else
        return (n*power(n,p-1));
}
float cos(int x, int n)
{
    float sum=0;
    for(int i=1;i<=n;i++)
        sum=sum+(power(-1,i)*power(x,2*i)/(fact(2*i)));
    return sum;
}
int main()
{
    int n,x;
    printf("Enter x and n");
    scanf("%d%d", &x, &n);
    int result=cos(x,n);
    printf("%d",result);

    return 0;
}

```

Enter x and n3

1

-4

Enter x and n7

2

200

Enter x and n4

7

0