

## ES202

### Assignment-IV

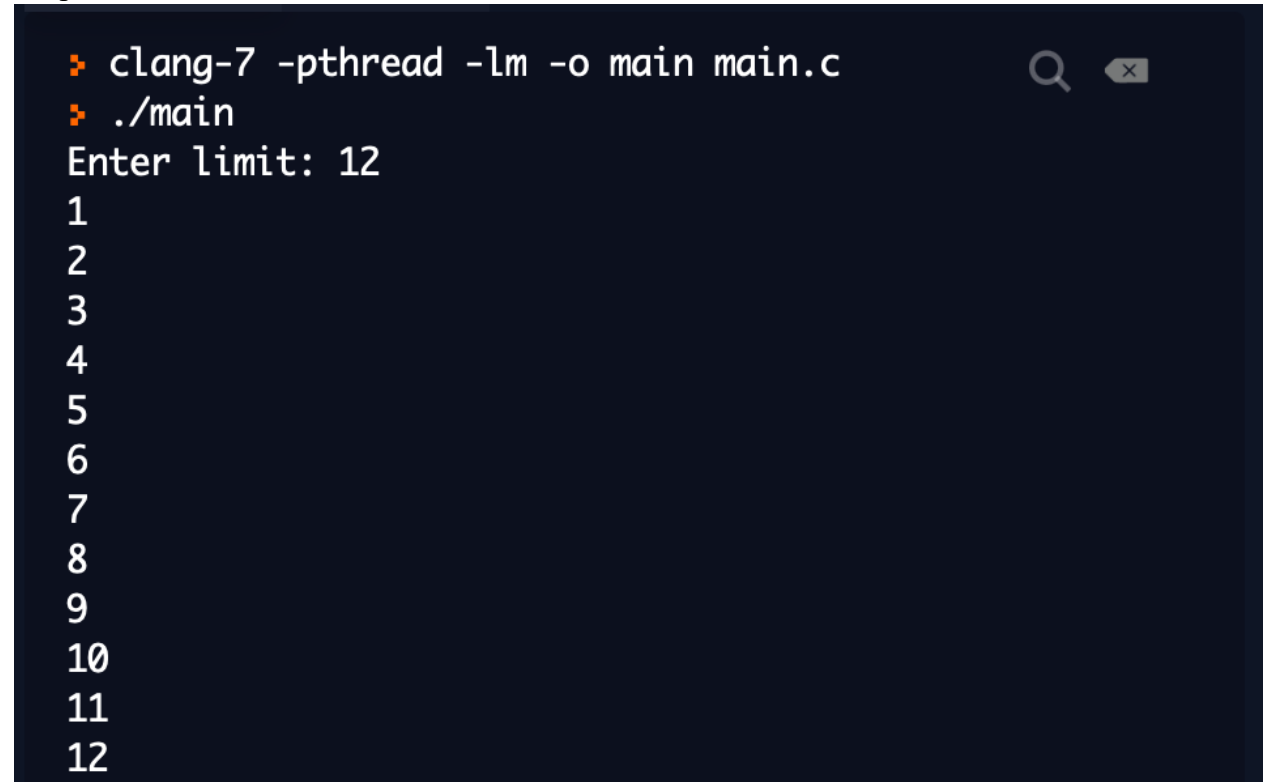
#### C Programming Exercises

1. Write a C program to print all-natural numbers between 1 to n using recursion.

Source Code:

```
#include <stdio.h>
int PrintNum(int n,int lim)
{
    if(n<=lim)
        printf("%d ",n);
    PrintNum(++n,lim);
}
void main()
{
    int l;
    printf("Enter limit: ");
    scanf("%d",&l);
    PrintNum(1,l);
}
```

Output:



```
> clang-7 -pthread -lm -o main main.c
> ./main
Enter limit: 12
1
2
3
4
5
6
7
8
9
10
11
12
```

2. Write a C program to print all even or odd numbers in given range using recursion.

Source Code:

```
#include <stdio.h>

void even(int n,int lim)
{
    if(n<=lim&& n%2==0)
    {
        printf("%d ",n);
        if(n%2==0)
            n+=2;
        else
            ++n;
        even(n,lim);
    }
}

void odd(int n,int lim)
{
    if(n<=lim&& n%2!=0)
    {
        printf("%d ",n);
        if(n%2!=0)
            n+=2;
        else
            ++n;
        odd(n,lim);
    }
}

int main()
{
    int s,e,c;
    printf("Enter range: ");
    scanf("%d %d",&s,&e);
    printf("Enter 1 for odd and 2 for even: ");
    scanf("%d",&c);
    if(c==1)
        odd(s,e);
```

```
    else if(c==2)
        even(s,e);
return 0;
}
```

Output:

```
❏ ./main
Enter range: 1
12
Enter 1 for odd and 2 for even: 1
1 3 5 7 9 11 exit status 174
❏
```

3. Write a C program to find sum of all-natural numbers between 1 to n using recursion.

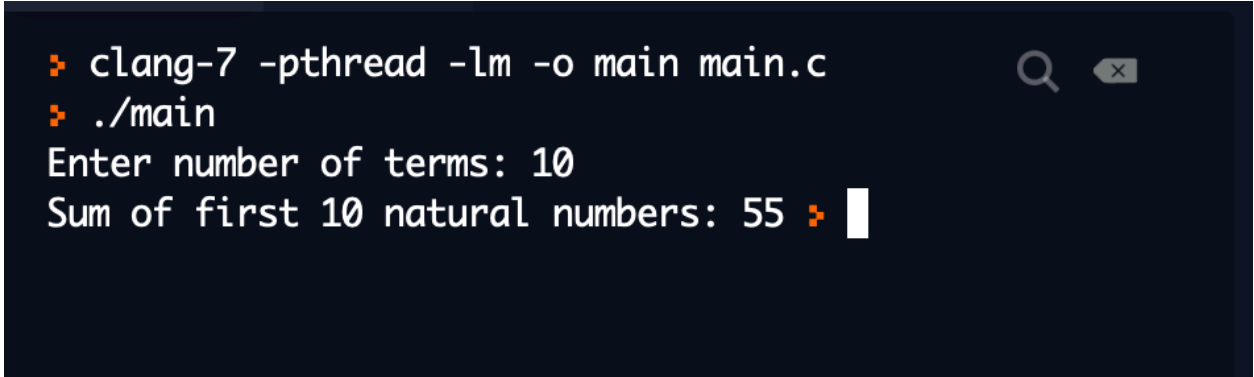
Source Code:

```
#include <stdio.h>

int Sum(int n)
{if (n != 0)
    return n + Sum(n - 1);
else
    return n;
}

int main()
{
    int l;
    printf("Enter number of terms: ");
    scanf("%d",&l);
    printf("Sum of first %d natural numbers: %d ",l,Sum(l));
    return 0;
}
```

Output:



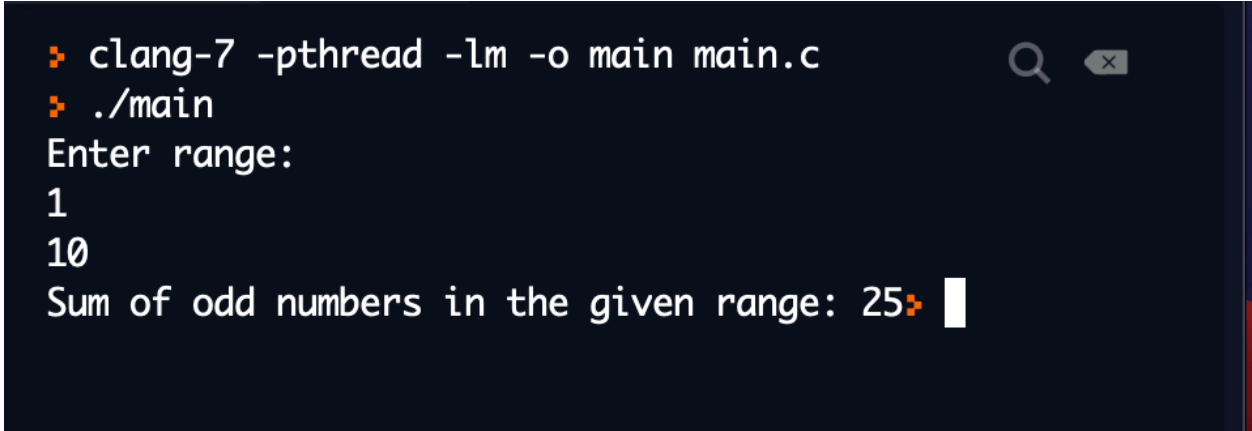
```
❏ clang-7 -pthread -lm -o main main.c
❏ ./main
Enter number of terms: 10
Sum of first 10 natural numbers: 55 ❏
```

4. Write a C program to find sum of all even or odd numbers in given range using recursion.

Source Code:

```
#include <stdio.h>
int sumOfEvenOdd(int s, int e)
{
    if(s > e)
        return 0;
    else
        return (s + sumOfEvenOdd(s + 2, e));
}
int main()
{
    int n1, n2;
    printf("Enter range: \n");
    scanf("%d",&n1);
    scanf("%d",&n2);
    if(n1%2==0)
        printf("Sum of even numbers in the given range: %d",sumOfEvenOdd(n1,n2));
    else
        printf("Sum of odd numbers in the given range: %d",sumOfEvenOdd(n1,n2));
    return 0;
}
```

Output:



```
❏ clang-7 -pthread -lm -o main main.c
❏ ./main
Enter range:
1
10
Sum of odd numbers in the given range: 25❏
```

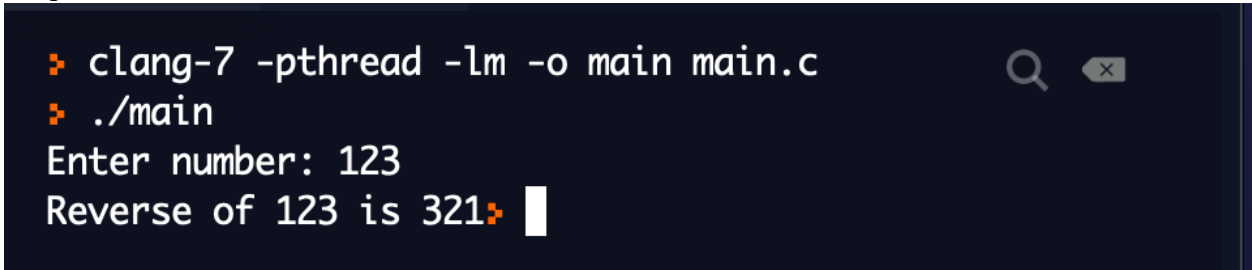
5. Write a C program to find reverse of any number using recursion.

Source Code:

```
#include <stdio.h>
int sum=0,rem;
int reverse(int n)
{
    if(n!=0)
    {
        rem=n%10;
        sum=sum*10+rem;
        reverse(n/10);
    }
    return sum;
}

int main()
{
    int n;
    printf("Enter number: ");
    scanf("%d",&n);
    printf("Reverse of %d is %d",n,reverse(n));
    return 0;
}
```

Output:



```
❯ clang-7 -pthread -lm -o main main.c
❯ ./main
Enter number: 123
Reverse of 123 is 321❯
```

The screenshot shows a terminal window with a dark background. It displays the compilation of the C program using 'clang-7' with flags '-pthread' and '-lm' to produce an executable named 'main'. After running './main', the program prompts for an input number. The user enters '123', and the program outputs 'Reverse of 123 is 321'. The prompt character is a red star-like symbol.

6. Write a C program to check whether a number is palindrome or not using recursion.

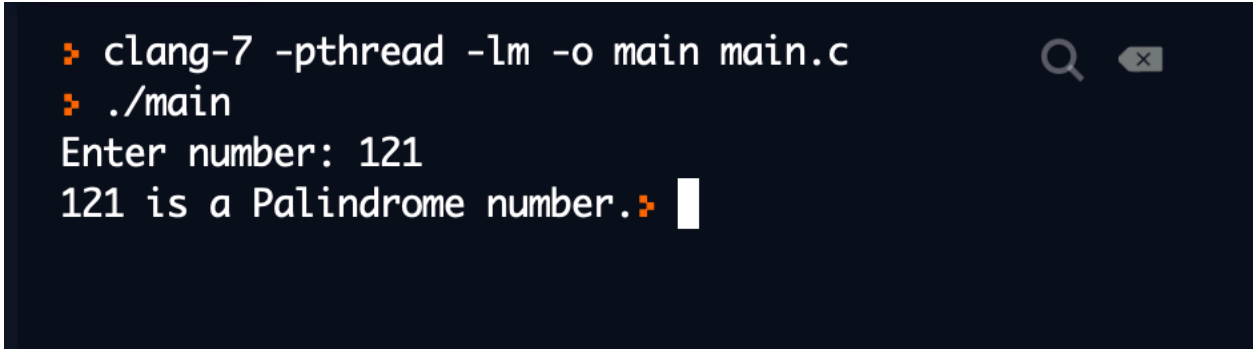
Source Code:

```
#include <stdio.h>
int sum=0,rem;
int reverse(int n)
{
    if(n!=0)
    {
        rem=n%10;
        sum=sum*10+rem;
        reverse(n/10);
    }
    return sum;
}

int main()
{
    int n;
    printf("Enter number: ");
    scanf("%d",&n);
    if(n==reverse(n))
        printf("%d is a Palindrome number.",n);
    else
        printf("%d is not a Palindrome number.",n);

    return 0;
}
```

Output:



```
❏ clang-7 -pthread -lm -o main main.c
❏ ./main
Enter number: 121
121 is a Palindrome number.❏
```

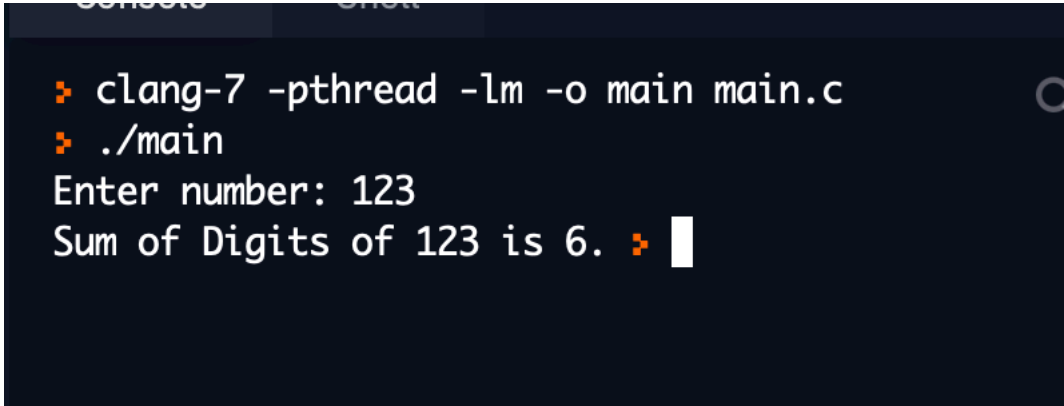
7. Write a C program to find sum of digits of a given number using recursion.

Source Code:

```
#include <stdio.h>
int sum=0;
int SoD(int n)
{
    if(n!=0)
    {
        sum+=n%10;
        SoD(n/10);
    }
    return sum;
}

void main()
{
    int n;
    printf("Enter number: ");
    scanf("%d",&n);
    printf("Sum of Digits of %d is %d. ",n,SoD(n));
return 0;
}
```

Output:



```
❯ clang-7 -pthread -lm -o main main.c
❯ ./main
Enter number: 123
Sum of Digits of 123 is 6. ❯
```



8. Write a C program to generate nth Fibonacci term using recursion.

Source Code:

```
#include <stdio.h>
int fibo(int num)
{
    if (num == 0)
    {
        return 0;
    }
    else if (num == 1)
    {
        return 1;
    }
    else
    {
        return(fibo(num - 1) + fibo(num - 2));
    }
}
void main()
{
    int n;
    printf("Enter which term to print from the Fibonacci Series: ");
    scanf("%d",&n);
    if (n < 0)
    {
        printf("Fibonacci of negative number is not possible.\n");
    }
    else
    {
        printf("The %d number in fibonacci series is %d\n", n, fibo(n-1));
    }
    return 0;
}
```

Output:

```
> clang-7 -pthread -lm -o main main.c
> ./main
Enter which term to print from the Fibonacci Series: 23
The 23 number in fibonacci series is 17711
> █
```

9. Write a C program to find GCD (HCF) of two numbers using recursion.

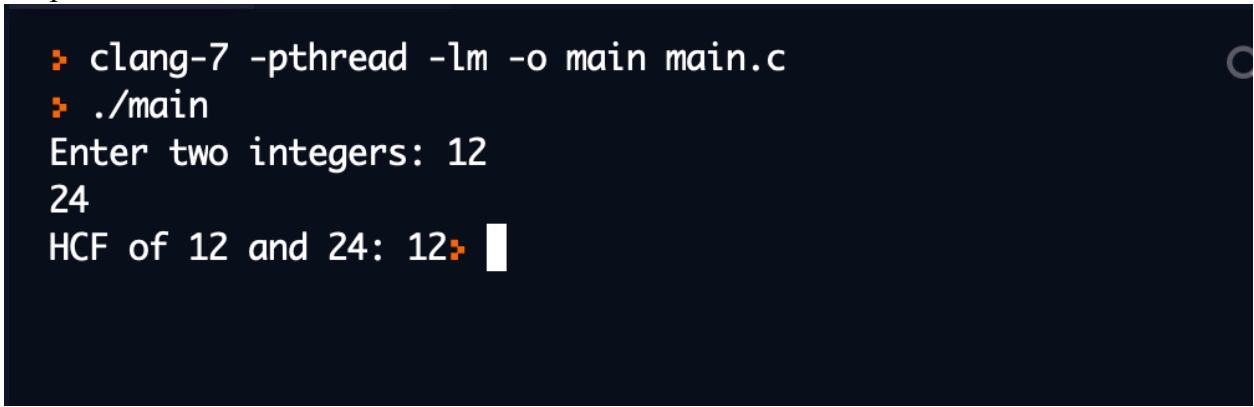
Source Code:

```
#include <stdio.h>
int hcf(int a,int b)
{
    if (b != 0)
        return hcf(b, a % b);
    else
        return a;
}
int main()
{
    int n1, n2;

    printf("Enter two integers: ");
    scanf("%d %d",&n1,&n2);

    n1 = ( n1 > 0 ) ? n1 : -n1;
    n2 = ( n2 > 0 ) ? n2 : -n2;
    printf("HCF of %d and %d: %d",n1,n2,hcf(n1,n2));
    return 0;
}
```

Output:



```
> clang-7 -pthread -lm -o main main.c
> ./main
Enter two integers: 12
24
HCF of 12 and 24: 12
```

10. Write a C program to find LCM of two numbers using recursion.

Source Code:

```
#include <stdio.h>

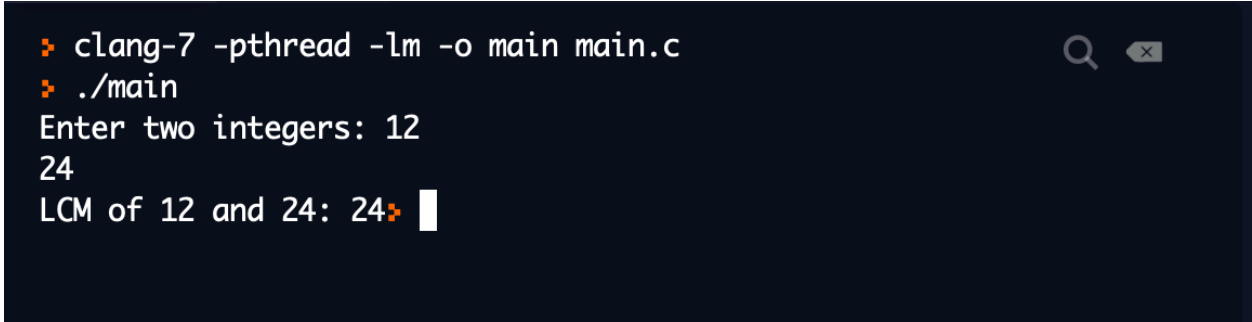
int lcm(int a, int b)
{
    static int c = 1;

    if (c % a == 0 && c % b == 0)
    {
        return c;
    }
    c++;
    lcm(a, b);
    return c;
}

int main()
{
    int n1, n2;

    printf("Enter two integers: ");
    scanf("%d %d", &n1, &n2);
    printf("LCM of %d and %d: %d", n1, n2, lcm(n1, n2));
    return 0;
}
```

Output:

A terminal window with a dark background. It shows the compilation of a C program using 'clang-7 -pthread -lm -o main main.c'. Then, it shows the execution of './main'. The program prompts 'Enter two integers: 12' and '24'. Finally, it outputs 'LCM of 12 and 24: 24' followed by a cursor.

```
❏ clang-7 -pthread -lm -o main main.c
❏ ./main
Enter two integers: 12
24
LCM of 12 and 24: 24❏
```