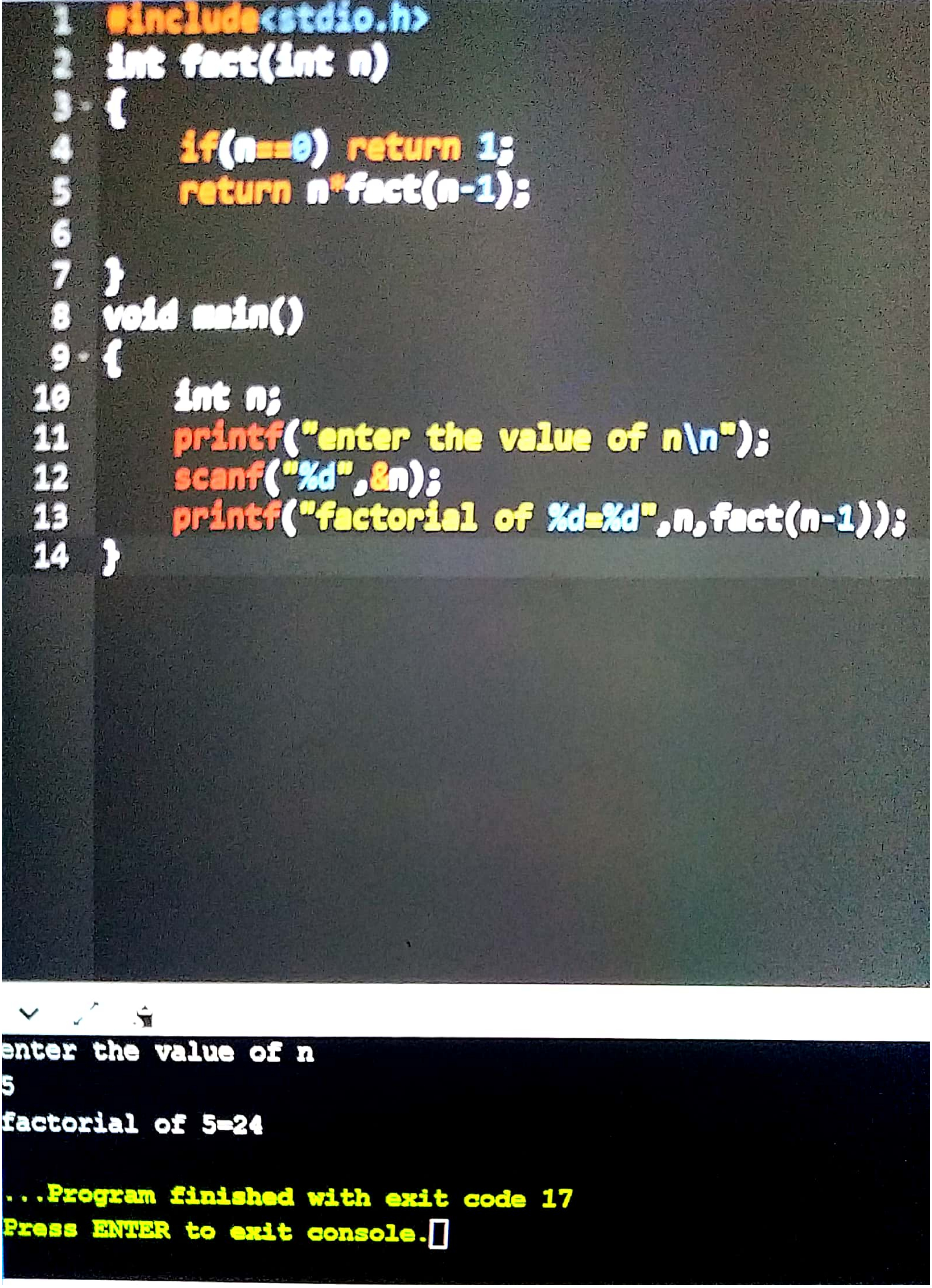
① Write a C-program to find factorial of 'n' using recursion

```
-6 # include < stdio.h> int fact (int n){
 if (n= =0) return 1;
 return  n* fact (n-1)
 }
Void main ()
{
int n ;
printf (" enter value of n\n");
scanf (" %d', &n);
printf (" factorial of %d = %d\n", n, fact(n)) ;
}.
```

② ~~W=prom~~ output :-

```
Enter value of 'n'
5
factorial of 5 = 24.
```



```c
#include<stdio.h>
int fact(int n)
{
        if(n==0) return 1;
        return n*fact(n-1);

}
void main()
{
       int n;
       printf("enter the value of n\n");
       scanf("%d",&n);
       printf("factorial of %d=%d",n,fact(n-1));
}
```

```
enter the value of n
5
factorial of 5=24

...Program finished with exit code 17
Press ENTER to exit console.
```

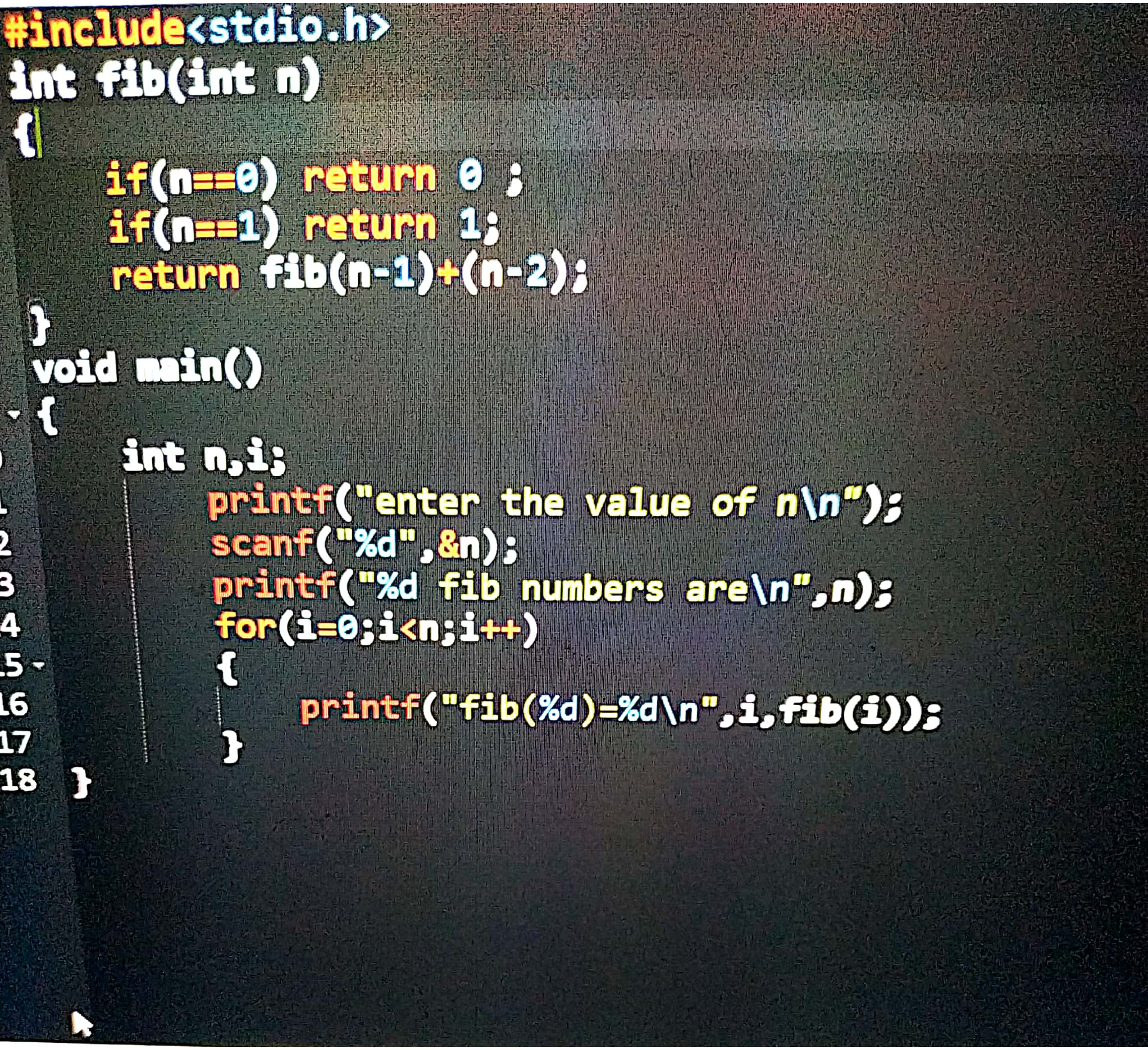
(2) W-C-program to print 'n' fibonacci series

```
:- # include< stdio.h>.
   int fib (int n)
   {
  if (n==0) return Ø;
  if (n==1) return 1 ;
   return fib (n-1) + fib (n-2);
   }
  void main( )
  {
```

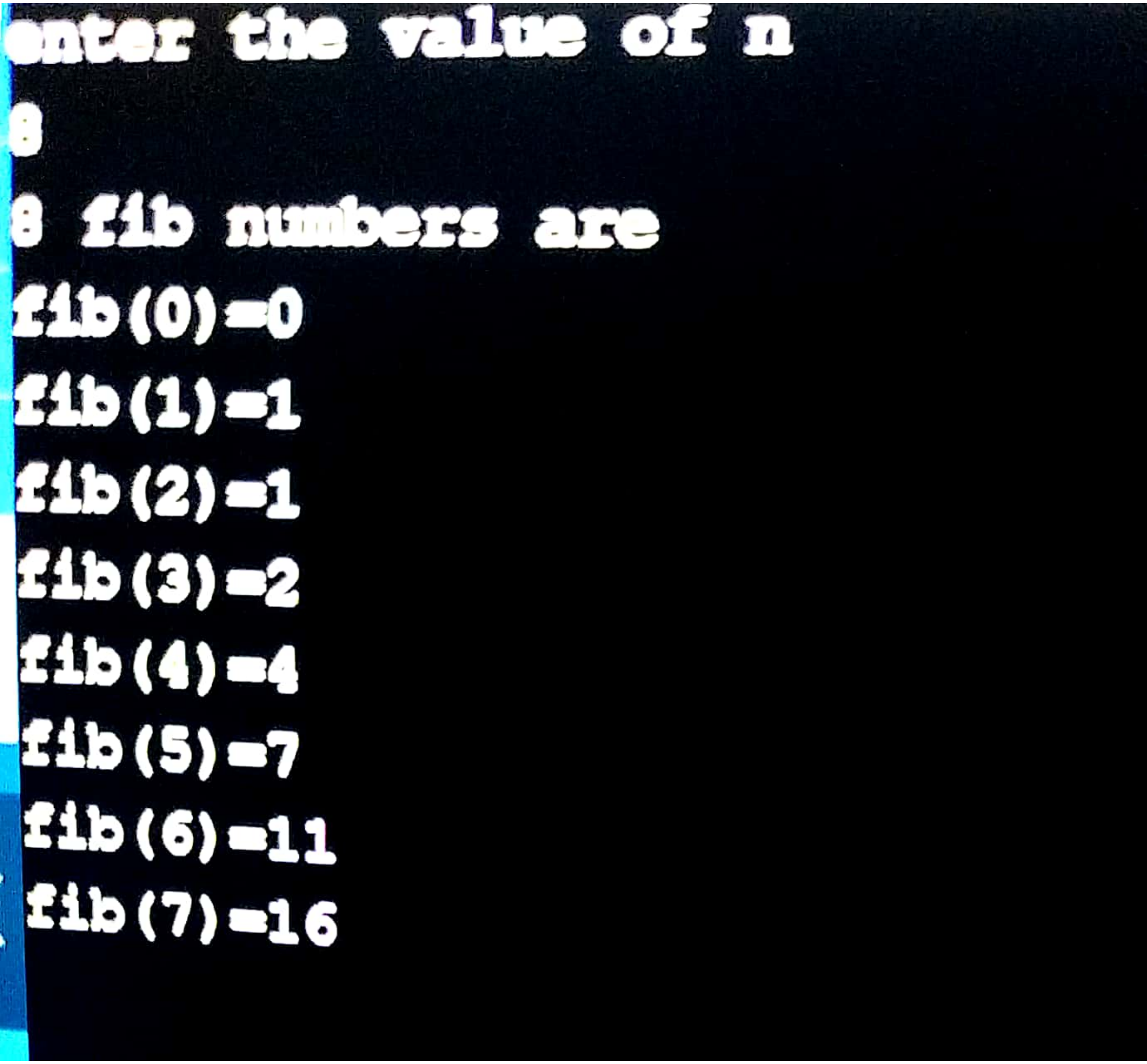
```
int i,n;
printf (" ente value of n\n");
scanf (" %d", &n);
printf ("%d fid numbers are\n",n);
for(i=0 ; i<n ;i++)
 {
printf (" fib(%d)=%d\n", i, fib(i));
}
```

Output :

```
enter the value of n
 8
8 fib numbers are
 fib(0) = 0
fib (1) = 1
fib (2) = 1
fib (3) = 2
fib(4) = 4
fib(5) = 7
fib(6) = 11
fib(7) = 16.
```



```c
#include<stdio.h>
int fib(int n)
{
    if(n==0) return 0 ;
    if(n==1) return 1;
    return fib(n-1)+(n-2);
}
void main()
{
    int n,i;
        printf("enter the value of n\n");
        scanf("%d",&n);
        printf("%d fib numbers are\n",n);
        for(i=0;i<n;i++)
        {
            printf("fib(%d)=%d\n",i,fib(i));
        }
}
```



```
enter the value of n
8
8 fib numbers are
fib(0)=0
fib(1)=1
fib(2)=1
fib(3)=2
fib(4)=4
fib(5)=7
fib(6)=11
fib(7)=16
```

3) Write a program to find GCD of 2-numbers using recursion.

=> ~~# include <stdio.h>~~

~~{~~

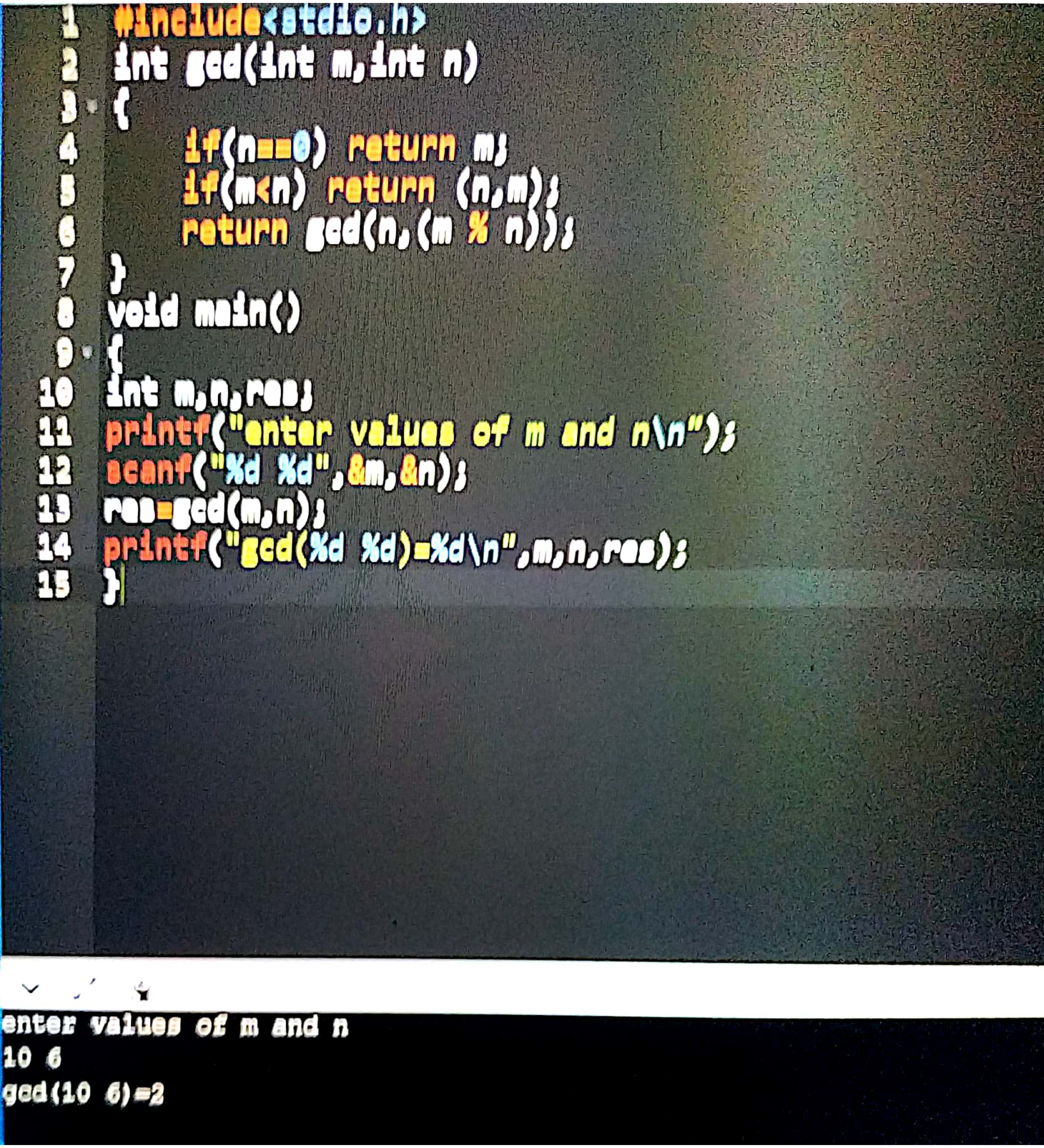
~~if (n= =0) return 0;~~

~~if (n= = 1) retu.~~

```
# include < stdio.h >
int gcd (int m, int n)
{
if (n = 0) return m;
if (m<n) return gcd (n,m);
return gcd (n,m%n);
}
void main()
{
int m,n,res;
 printf ("enter value of m & n\n")
 scanf ("%d %d",&m,&n);
res = gcd (m,n).
printf ("gcd (%d,%d) = %d\n", m,n,r
 }
```

output :

```
Enter the values of m and n
 10 6
gcd (10 6) = 2.
```

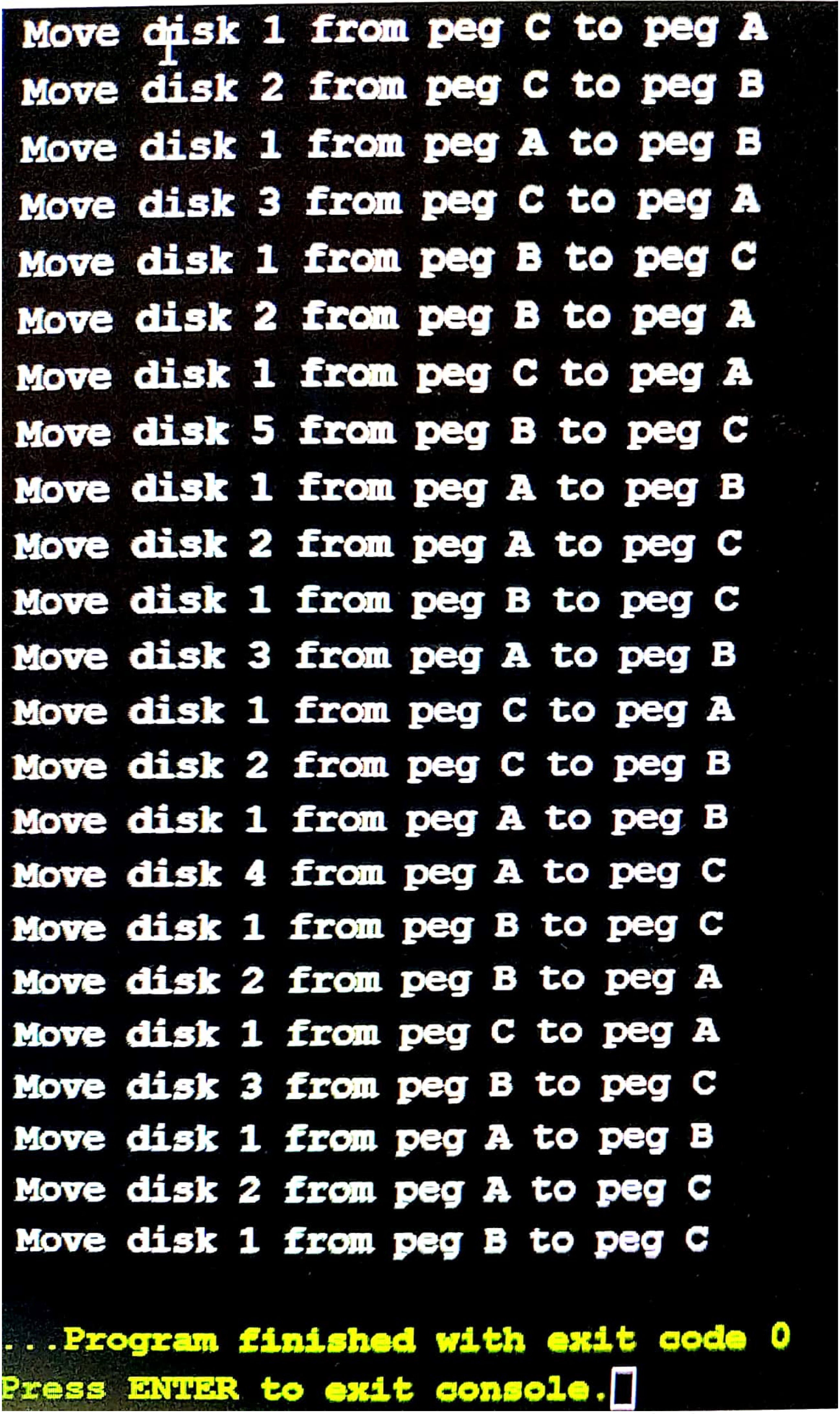


```c
#include<stdio.h>
int gcd(int m,int n)
{
        if(n==0) return m;
        if(m<n) return (n,m);
        return gcd(n,(m % n));
}
void main()
{
int m,n,res;
printf("enter values of m and n\n");
scanf("%d %d",&m,&n);
res=gcd(m,n);
printf("gcd(%d %d)=%d\n",m,n,res);
}
```

```
enter values of m and n
10 6
gcd(10 6)=2
```

4. Tower of Hanoi

```
# include<stdio.h>
void tower (int n, char source, char temp,
                    char dest)
{
if (n==1)
 {
printf ("move des 1 from %c to %c\n",
                 source, dest);
return ; }
}
towers(n-1, source, dest, temp);
printf (" mov disk ~~from~~ %d from
              source to dest\n", n);
towers (n-1, temp, dest);
}
main ()
{
int n;
printf(" enter no. of disks\n");
scanf("%d",&n);
Towers (n, 'S', 'T', 'D');
}
```



```
Move disk 1 from peg C to peg A
Move disk 2 from peg C to peg B
Move disk 1 from peg A to peg B
Move disk 3 from peg C to peg A
Move disk 1 from peg B to peg C
Move disk 2 from peg B to peg A
Move disk 1 from peg C to peg A
Move disk 5 from peg B to peg C
Move disk 1 from peg A to peg B
Move disk 2 from peg A to peg C
Move disk 1 from peg B to peg C
Move disk 3 from peg A to peg B
Move disk 1 from peg C to peg A
Move disk 2 from peg C to peg B
Move disk 1 from peg A to peg B
Move disk 4 from peg A to peg C
Move disk 1 from peg B to peg C
Move disk 2 from peg B to peg A
Move disk 1 from peg C to peg A
Move disk 3 from peg B to peg C
Move disk 1 from peg A to peg B
Move disk 2 from peg A to peg C
Move disk 1 from peg B to peg C


...Program finished with exit code 0
Press ENTER to exit console.
```