

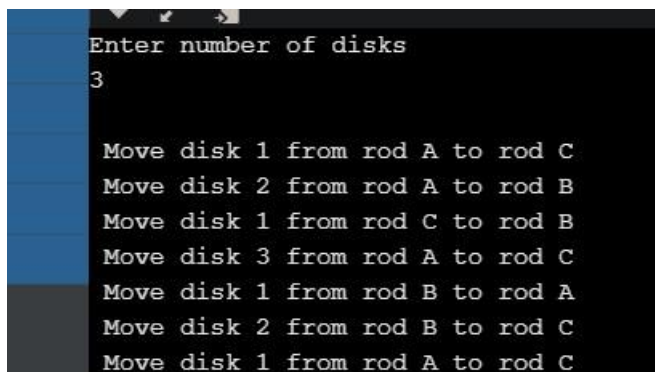
TowerOfhanoi :

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

```
#include<stdlib.h>
```

```
void towerOfHanoi(int n, char from_rod, char to_rod, char temp_rod)
{
    if (n == 1)
    {
        printf("\n Move disk 1 from rod %c to rod %c", from_rod, to_rod);
        return;
    }
    towerOfHanoi(n-1, from_rod, temp_rod, to_rod);
    printf("\n Move disk %d from rod %c to rod %c", n, from_rod, to_rod);
    towerOfHanoi(n-1, temp_rod, to_rod, from_rod);
}
```

```
int main()
{ int n;
  printf("Enter number of disks\n");
  scanf("%d",&n);
  towerOfHanoi(n, 'A', 'C', 'B');
  return 0;
}
```



```
Enter number of disks
3
Move disk 1 from rod A to rod C
Move disk 2 from rod A to rod B
Move disk 1 from rod C to rod B
Move disk 3 from rod A to rod C
Move disk 1 from rod B to rod A
Move disk 2 from rod B to rod C
Move disk 1 from rod A to rod C
```

Binary Search :

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

```
int binarySearch(int arr[], int l, int r, int x)
```

```
{
```

```
    if (r >= l)
```

```
    {
```

```
        int mid = l + (r - l)/2;
```

```
        if (arr[mid] == x) return mid;    if (arr[mid] > x) return binarySearch(arr, l, mid-1, x);
```

```
        return binarySearch(arr, mid+1, r, x);
```

```
    }
```

```
    return -1;
```

```
}
```

```
int main(void)
```

```
{
```

```
    int arr[50];
```

```
    int n,i,x;
```

```
    printf("Enter the size of array\n");
```

```
    scanf("%d",&n);
```

```
    for(i=0;i<n;i++)
```

```
    { printf("Enter the %d element\n",i+1);
```

```
        scanf(" %d",&arr[i]);}
```

```
    printf("Enter the number to search\n");
```

```
    scanf(" %d",&x);
```

```
    int result = binarySearch(arr, 0, n-1, x);
```

```
    (result == -1)? printf("Element is not present in array")
```

```
        : printf("Element is present at index %d", result);
```

```
    return 0;
```

```
}
```

```
Enter the size of array
5
Enter the 1 element
3
Enter the 2 element
4
Enter the 3 element
8
Enter the 4 element
10
Enter the 5 element
15
Enter the number to search
10
Element is present at index 3
```

Fibinocii :

```
#include <stdlib.h>
```

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

```
int fib(int n)
```

```
{
```

```
    if (n <= 1)
```

```
        return n;
```

```
    return fib(n-1) + fib(n-2);
```

```
}
```

```
int main ()
```

```
{
```

```
    int n;
```

```
    printf("Enter number of terms\n");
```

```
    scanf("%d",&n);
```

```
    printf("%d", fib(n));
```

```
    return 0;
```

```
}
```

```
Enter number of terms
10
55
```

Factorial :

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

```
#include<stdlib.h>
```

```
int fact(int n);
```

```
int main() {
```

```
    int n;
```

```
    printf("Enter a positive integer: ");
```

```
    scanf("%d",&n);
```

```
    while(n<0){
```

```
        printf("Enter positive number\n");
```

```
        scanf("%d",&n);}
```

```
    printf("Factorial of %d = %d", n, fact(n));
```

```
    return 0;
```

```
}
```

```
int fact(int n) {
```

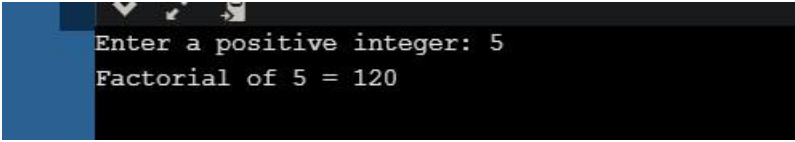
```
    if (n>=1)
```

```
        return n*fact(n-1);
```

```
    else
```

```
        return 1;
```

```
}
```

A terminal window with a dark background and a blue vertical bar on the left. It shows the prompt "Enter a positive integer: 5" and the output "Factorial of 5 = 120".

```
Enter a positive integer: 5
Factorial of 5 = 120
```

GCD:

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

```
#include<stdlib.h>
```

```
int gcd(int n1, int n2);
```

```
int main() {
```

```
    int n1, n2;
```

```
    printf("Enter two integers: ");
```

```
    scanf("%d %d", &n1, &n2);
```

```
    printf("G.C.D of %d and %d is %d.", n1, n2, gcd(n1, n2));
```

```
    return 0;
```

```
}
```

```
int gcd(int n1, int n2) {
```

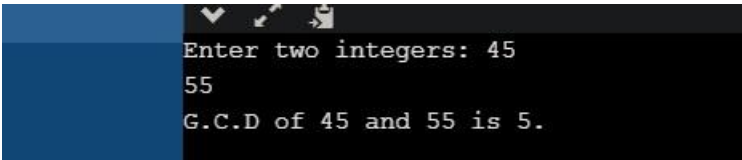
```
    if (n2 != 0)
```

```
        return gcd(n2, n1 % n2);
```

```
    else
```

```
        return n1;
```

```
}
```

A terminal window with a dark background and a blue vertical bar on the left. It shows the prompt "Enter two integers: 45", the input "55", and the output "G.C.D of 45 and 55 is 5.". The input "55" is on a new line.

```
Enter two integers: 45
55
G.C.D of 45 and 55 is 5.
```