

```
#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[] = { 2, 3, 4, 10, 40 };
```

```
    int n = sizeof(arr) / sizeof(arr[0]);
```

```
    int x = 10;
```

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

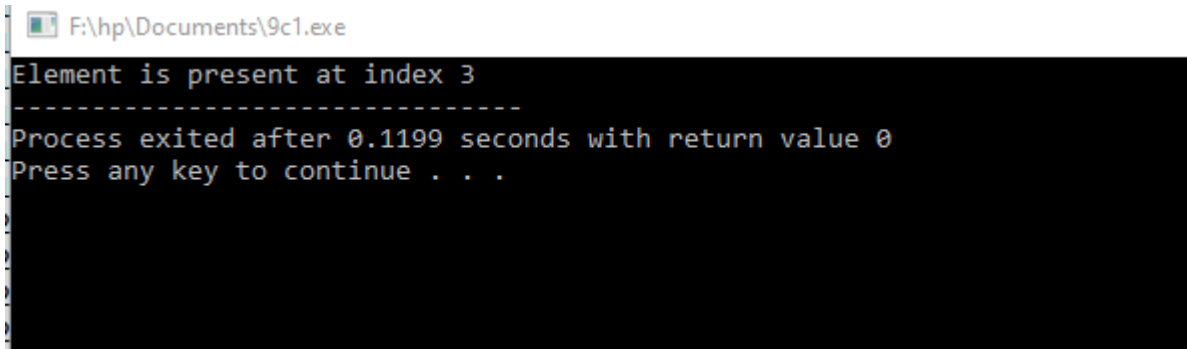
```
    (result == -1) ? printf("Element is not present in array")
```

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

```
        result);

    return 0;
}
```

OUTPUT:



```
F:\hp\Documents\9c1.exe
Element is present at index 3
-----
Process exited after 0.1199 seconds with return value 0
Press any key to continue . . .
```

TOWER OF HANOI:

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

```
void towerOfHanoi(int n, char from_rod, char to_rod, char aux_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, aux_rod, to_rod);
    printf("\n Move disk %d from rod %c to rod %c", n, from_rod, to_rod);
    towerOfHanoi(n-1, aux_rod, to_rod, from_rod);
}
```

```
int main()
{
    int n;

    printf("Enter the number of disc");

    scanf("%d",&n);

    towerOfHanoi(n, 'A', 'C', 'B');
```

```
    return 0;
}
```

OUTPUT:

```
Enter the number of disc4

Move disk 1 from rod A to rod B
Move disk 2 from rod A to rod C
Move disk 1 from rod B to rod C
Move disk 3 from rod A to rod B
Move disk 1 from rod C to rod A
Move disk 2 from rod C to rod B
Move disk 1 from rod A to rod B
Move disk 4 from rod A to rod C
Move disk 1 from rod B to rod C
Move disk 2 from rod B to rod A
Move disk 1 from rod C to rod A
Move disk 3 from rod B to rod C
Move disk 1 from rod A to rod B
Move disk 2 from rod A to rod C
Move disk 1 from rod B to rod C
-----
Process exited after 1.55 seconds with return value 0
Press any key to continue . . .
```

GCD OF TWO NUMBERS:

```
#include <stdio.h>

int main()
{
    int n1, n2, i, gcd;

    printf("Enter two integers: ");

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

    for(i=1; i <= n1 && i <= n2; ++i)
    {

        if(n1%i==0 && n2%i==0)

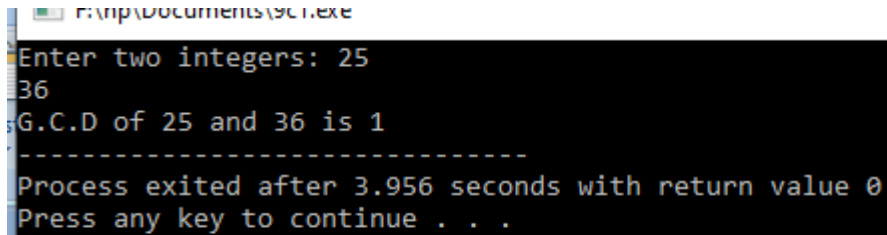
            gcd = i;
    }
}
```

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

```
return 0;
```

```
}
```

OUTPUT:



```
C:\np\Documents\gcd.exe
Enter two integers: 25
36
G.C.D of 25 and 36 is 1
-----
Process exited after 3.956 seconds with return value 0
Press any key to continue . . .
```

FATORIAL:

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

```
int main() {
```

```
    int n, i;
```

```
    unsigned long long fact = 1;
```

```
    printf("Enter an integer: ");
```

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

```
    if (n < 0)
```

```
        printf("Error! Factorial of a negative number doesn't exist.");
```

```
    else {
```

```
        for (i = 1; i <= n; ++i) {
```

```
            fact *= i;
```

```
        }
```

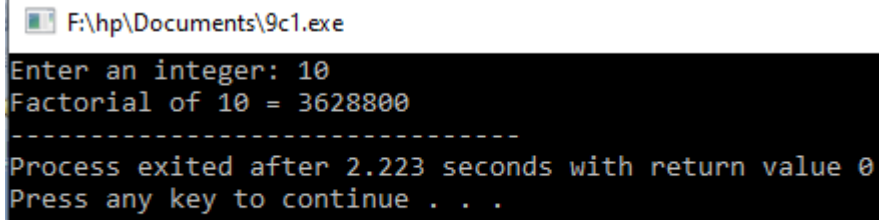
```
        printf("Factorial of %d = %llu", n, fact);
```

```
    }
```

```
    return 0;
```

```
}
```

OUTPUT:



```
F:\hp\Documents\9c1.exe
Enter an integer: 10
Factorial of 10 = 3628800
-----
Process exited after 2.223 seconds with return value 0
Press any key to continue . . .
```

Nth FIBONACCI:

```
#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 the term: ");
```

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

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

```
getchar();
```

```
return 0;
```

```
}
```

OUTPUT:

F:\hp\Documents\9c1.exe

Enter the term: 10

55

-----

Process exited after 2.023 seconds with return value 0

Press any key to continue . . .