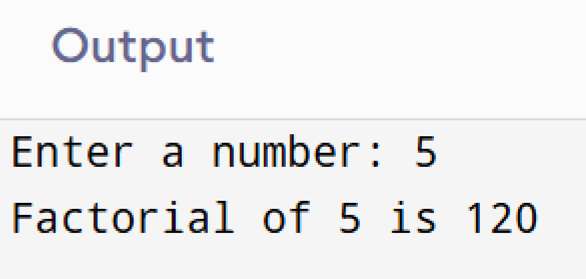
1. Write a function to find the factorial of a number.

IPO
Input:Enter a value as input.
Proces:To find the factorial of a number.
Output: output he variable.

Program:

```
#include <stdio.h>

// Function to calculate factorial
int factorial(int n) {
    int fact = 1;
    for(int i = 1; i <= n; i++) {
        fact *= i;
    }
    return fact;
}

int main() {
    int number;
    printf("Enter a number: ");
    scanf("%d", &number);

    if(number < 0) {
        printf("Factorial is not defined for negative numbers.\n");
    } else {
        printf("Factorial of %d is %d\n", number, factorial(number));
    }

    return 0;
}
```

Output



```
Enter a number: 5
Factorial of 5 is 120
```

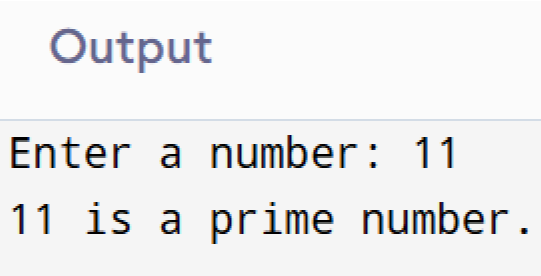
2. Write a function to check whether a number is prime.

IPO
Input:Enter a value as input.
Proces:To check whether a number is prime.
Output: output he variable.

Program:

```
#include <stdio.h>
int isPrime(int n)
{
   if (n <= 1) return 0;
   for (int i = 2; i < n; i++) {
      if (n % i == 0) return 0;
   }
   return 1;
}
int main() {
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);
   if (isPrime(num))
      printf("%d is a prime number.\n", num);
   else
      printf("%d is not a prime number.\n", num);

   return 0;
}
```

Output



Output

```
Enter a number: 11
11 is a prime number.
```

3. Write a function to calculate power using recursion.

IPO
Input:Enter a value as input.
Proces:To calculate power using recursion.
Output: output he variable.

Program

```
#include <stdio.h>
int power(int base, int exponent)
 {
    if (exponent == 0)
        return 1;
    else
        return base * power(base, exponent - 1);
}
int main() {
    int base, exponent;
    printf("Enter base and exponent: ");
    scanf("%d %d", &base, &exponent);
    printf("%d^%d = %d\n", base, exponent, power(base, exponent));
    return 0;
}
```

Output

```
Enter base and exponent: 2 3
2^3 = 8
```

4. Write a function to check palindrome number using recursion.

IPO
Input:Enter a value as input.
Proces:To calculate power using recursion.
Output: output he variable.

Program

```
#include <stdio.h>

// Recursive function to reverse the number
```

```
int reverse(int num, int rev) {
    if (num == 0)
        return rev;
    return reverse(num / 10, rev * 10 + num % 10);
}

// Function to check if number is palindrome
int isPalindrome(int num) {
    return num == reverse(num, 0);
}
int main()
{
    int number;
    printf("Enter a number: ");
    scanf("%d", &number);
    if (isPalindrome(number))
        printf("%d is a palindrome number.\n", number);
    else
        printf("%d is not a palindrome number.\n", number);
    return 0;
}
```

Output

```
Enter a number: 121
121 is a palindrome number.
```

5. Write a function to calculate nCr (combinations).

IPO
Input:Enter a value as input.
Proces:To calculate nCr (combinations).
Output: output he variable.

Program

```
#include <stdio.h>
int factorial(int n) {
    int fact = 1;
    for(int i = 1; i <= n; i++)
```

```
        fact *= i;
    return fact;
}
int nCr(int n, int r) {
    return factorial(n) / (factorial(r) * factorial(n - r));
}
int main()
{
    int n, r;
    printf("Enter n and r: ");
    scanf("%d %d", &n, &r);
    if (r > n)
        printf("Invalid input: r cannot be greater than n.\n");
    else
        printf("nCr = %d\n", nCr(n, r));
    return 0;
}
```

Output

```
Enter n and r: 5 2
nCr = 10
```

6. Write a program to demonstrate call by value and call by reference.

IPO

Input:Enter a value as input.
Proces:To demonstrate call by value and call by reference.
Output: output he variable.

Program

```
#include <stdio.h>
void callByValue(int a) {
    a = a + 10;
    printf("Inside callByValue: a = %d\n", a);
}
void callByReference(int *b) {
    *b = *b + 10;
    printf("Inside callByReference: b = %d\n", *b);
}
int main()
{
    int x = 5, y = 5;
    printf("Before callByValue: x = %d\n", x);
    callByValue(x);
    printf("After callByValue: x = %d\n\n", x);
    printf("Before callByReference: y = %d\n", y);
    callByReference(&y);
    printf("After callByReference: y = %d\n", y);
    return 0;
}
```

Output

```
Before callByValue: x = 5
Inside callByValue: a = 15
After callByValue: x = 5

Before callByReference: y = 5
Inside callByReference: b = 15
After callByReference: y = 15
```

7. Write a program using function to swap two numbers.

IPO

Input:Enter a value as input.
Proces:To swap two numbers.
Output: output he variable.

Program

```
#include <stdio.h>
void swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int x, y;
    printf("Enter two numbers: ");
    scanf("%d %d", &x, &y);
    printf("Before swapping: x = %d, y = %d\n", x, y);
    swap(&x, &y);
    printf("After swapping: x = %d, y = %d\n", x, y);
    return 0;
}
```

Output,

```
Enter two numbers: 3 7
Before swapping: x = 3, y = 7
After swapping: x = 7, y = 3
```

8. Write a recursive function to find the nth Fibonacci number.

IPO
Input:Enter a value as input.
Proces:To find the nth Fibonacci number.
Output: output he variable.

Program

```
#include <stdio.h>
int fibonacci(int n) {
    if (n == 0)
        return 0;
    else if (n == 1)
        return 1;
    else
        return fibonacci(n - 1) + fibonacci(n - 2);
}
int main() {
    int n;
    printf("Enter n: ");
    scanf("%d", &n);
    printf("Fibonacci number at position %d is %d\n", n, fibonacci(n));
    return 0;
}
```

Output

```
Enter n: 6
Fibonacci number at position 6 is 8
```

9. Write a program to find GCD and LCM using functions.

IPO

Input:Enter a value as input.

Proces:To find GCD and LCM using functions..

Output: output he variable.

Program

```
#include <stdio.h>
int findGCD(int a, int b
) {
   while (b != 0)
{
      int temp = b;
      b = a % b;
      a = temp;
   }
   return a;
}
int findLCM(int a, int b)
 {
   return (a * b) / findGCD(a, b);
}
int main()
{
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   int gcd = findGCD(num1, num2);
   int lcm = findLCM(num1, num2);
   printf("GCD of %d and %d is %d\n", num1, num2, gcd);
   printf("LCM of %d and %d is %d\n", num1, num2, lcm);
   return 0;
}
```

Output

```
Enter two numbers: 12 18
GCD of 12 and 18 is 6
LCM of 12 and 18 is 36
```

10. Write a program to demonstrate global and local variables.

IPO
Input:Enter a value as input.
Proces:To demonstrate global and local variables.
Output: output he variable.

Program

```c
#include <stdio.h>

// Global variable
int globalVar = 100;

void showVariables() {
    // Local variable
    int localVar = 50;

    printf("Inside function:\n");
    printf("Global variable = %d\n", globalVar);
    printf("Local variable = %d\n", localVar);
}

int main() {
    // Local variable
    int localVar = 10;

    printf("Inside main:\n");
    printf("Global variable = %d\n", globalVar);
    printf("Local variable = %d\n", localVar);

    showVariables();
return 0;
}
```

Output

```
Inside main:
Global variable = 100
Local variable = 10
Inside function:
Global variable = 100
Local variable = 50
```