Basic Programs on Functions

1. Sum of Two Numbers:

Write a program that defines a function to calculate the sum of two numbers and then calls the function to display the result.

```
#include <stdio.h>

int add(int a, int b) {
   return a + b;
}

int main() {
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   int result = add(num1, num2);
   printf("Sum: %d\n", result);
   return 0;
}
```

2. Factorial of a Number:

Write a program to calculate the factorial of a number using a function.

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

int main() {
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);
   int result = factorial(num);
   printf("Factorial: %d\n", result);
   return 0;
}
```

3. Check for Prime Number:

Create a program that checks whether a given number is prime or not using a function.

```
#include <stdio.h>
int isPrime(int n) {
   if (n <= 1) return 0; // 0 and 1 are not prime numbers
   for (int i = 2; i <= n / 2; i++) {
      if (n % i == 0)
         return 0; // If divisible, then not a prime number
   }
   return 1; // If no divisors were found, it's prime
}

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;
}
```

4. Swap Two Numbers:

Write a program that swaps the values of two variables using a function.

```
#include <stdio.h>
// Function to swap two numbers without using pointers
void swap(int a, int b) {
   int temp = a;
   a = b;
   b = temp;
   printf("Inside swap function: num1 = %d, num2 = %d\n", a, b);
}
int main() {
   int num1, num2;
   // Take input from the user
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   // Print the values before swapping
   printf("Before swap: num1 = %d, num2 = %d\n", num1, num2);
   // Call the swap function
   swap(num1, num2);
   // Print the values after swapping in main (will remain unchanged)
   printf("After swap in main: num1 = %d, num2 = %d\n", num1, num2);
   return 0;
}
```

5. Calculate the GCD (Greatest Common Divisor):

```
#include <stdio.h>
int gcd(int a, int b) {
   while (b != 0) {
      int temp = b;
      b = a % b;
      a = temp;
   }
   return a;
}

int main() {
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   int result = gcd(num1, num2);
   printf("GCD of %d and %d is %d\n", num1, num2, result);
   return 0;
}
```

Programs Based on Recursion:

1. Sum of Two Numbers (Using Recursion):

```
#include <stdio.h>
int add(int a, int b) {
   if (b == 0)
      return a;
   return add(a + 1, b - 1); // increment a and decrement b until b is zero
}
int main() {
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   int result = add(num1, num2);
   printf("Sum: %d\n", result);
   return 0;
}
```

2. Factorial of a Number (Using Recursion):

```
#include <stdio.h>
int factorial(int n) {
   if (n <= 1)
      return 1; // base case
   return n * factorial(n - 1); // recursive case
}
int main() {
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);
   int result = factorial(num);
   printf("Factorial: %d\n", result);
   return 0;
}
```

3. Calculate the GCD (Greatest Common Divisor – With Recursion):

```
#include <stdio.h>
int gcd(int a, int b) {
   if (b == 0)
      return a;
   return gcd(b, a % b);
}

int main() {
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   int result = gcd(num1, num2);
   printf("GCD of %d and %d is %d\n", num1, num2, result);
   return 0;
}
```

4. Swap 2 Numbers (Using Recursion)

```
#include <stdio.h>
void swapRecursive(int *a, int *b, int step) {
   if (step == 0) return;

   // Swapping step by changing values recursively
   int temp = *a;
   *a = *b;
   *b = temp;
   swapRecursive(a, b, step - 1);
}

int main() {
   int num1, num2;
   printf("Enter two numbers: ");
   scanf("%d %d", &num1, &num2);
   printf("Before swap: num1 = %d, num2 = %d\n", num1, num2);

   swapRecursive(&num1, &num2, 1);

   printf("After swap: num1 = %d, num2 = %d\n", num1, num2);
   return 0;
}
```