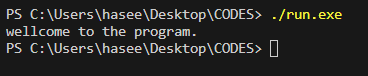
**Program #1: Display a simple message using the function.**

|  |
| --- |
| #include <iostream>  using namespace std;  void display\_message()  {  cout << "wellcome to the program.";  }  int main()  {  display\_message();  return 0;  } |

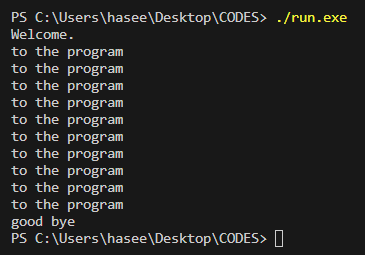
Output



**Program # 2: Display a simple message using the function and call a function using for loop.**

|  |
| --- |
| #include <iostream>  using namespace std;  void display\_message()  {  cout << "Welcome.\n";  for (int i = 0; i < 10; i++)  {  cout << "to the program"<<endl;  }  cout << "good bye";  }  int main()  {  display\_message();  return 0;  } |

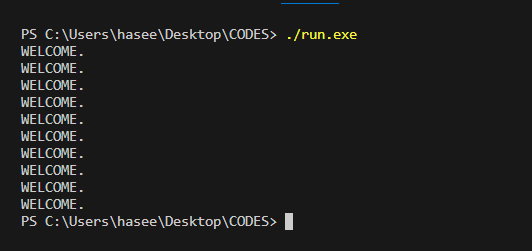
Output



**Program # 3: Display a simple message using the function and call a function using while loop.**

|  |
| --- |
| #include <iostream>  using namespace std;  void display\_message()  {  cout << "WELCOME."<<endl;  }  int main()  {  int message = 0;  while (message < 10)  {  display\_message();  message++;  }  return 0;  } |

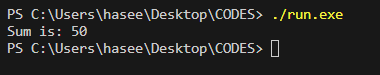
Output



**Program # 4: Write program to show the one number by using function.**

|  |
| --- |
| #include <iostream>  using namespace std;  int sum(int num, int num2)  {    int sum\_of\_nums = num + num2;  return sum\_of\_nums;  }  int main()  {  int result = sum(10,40);  cout << "Sum is: " << result;  return 0;  } |

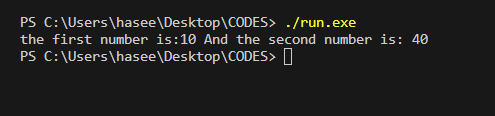
Output



**Program # 5: Write program to show the one number by using function.**

|  |
| --- |
| #include <iostream>  using namespace std;  void number(int num, int num2)  {  cout << "the first number is:" << num << " And the second number is: "<<num2;    }  int main()  {  number(10,40);  return 0;  } |

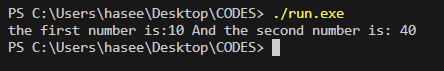
Output



**Program #6:**

|  |
| --- |
| #include <iostream>  using namespace std;  void number(int num, int num2);  int main()  {  number(10,40);  return 0;  }  void number(int num, int num2)  {    cout << "the first number is:" << num << " And the second number is: "<<num2;    } |

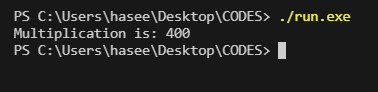
Output



**Program #7: Write a program to show one number by using the function Prototype.**

|  |
| --- |
| #include <iostream>  using namespace std;  int multiply(int num, int num2);  int main()  {  int result = multiply(10,40);  cout << "Multiplication is: " << result;  return 0;  }  int multiply(int num, int num2)  {    int sum\_of\_nums = num \* num2;  return sum\_of\_nums;  } |

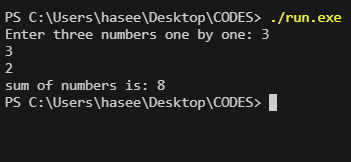
Ouput



**Program #8: Write a program to execute the sum of three numbers by using a Function prototype.**

|  |
| --- |
| #include <iostream>  using namespace std;  void sum(int num1, int num2, int num3);  int main()  {  int num1, num2 , num3;  cout << "Enter three numbers one by one: ";  cin >> num1 >> num2 >> num3;  sum(num1, num2, num3);    return 0;  }  void sum(int num1, int num2, int num3)  {  cout << "sum of numbers is: " << (num1 + num2 +num3);  } |

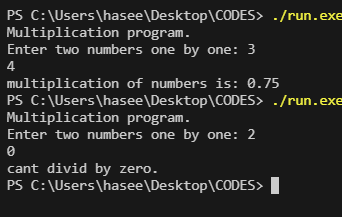
Output



**Program #9: Write a program to divide the two numbers by using Functions and check that if second number is zero then the program is terminated.**

|  |
| --- |
| #include <iostream>  using namespace std;  void division(double num1, double num2);  int main()  {  int num1, num2;  cout << "Multiplication program.\n";  cout << "Enter two numbers one by one: ";  cin >> num1 >> num2 ;  division(num1, num2);    return 0;  }  void division(double num1, double num2)  {  if (num2 == 0)  {  cout << "cant divid by zero.";  }  else  {  cout << "multiplication of numbers is: " << (num1 / num2 );  }  } |

Output

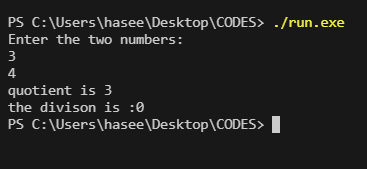


**Program #10:**

**Write a program that contains a function that takes two arguments of integer types and displays their quotient.**

|  |
| --- |
| #include <iostream>  using namespace std;  void quotient(int num1, int num2)  {  cout << "quotient is " << num1<<endl;  if (num2 == 0)  {  cout << "not possible.";  }else  {  cout << "the divison is :" << (num1 / num2) ;  }    }  int main()  {  int num1 ,num2;  cout << "Enter the two numbers:"<<endl;  cin >> num1 >> num2;  quotient(num1 , num2);    return 0;  } |

Output

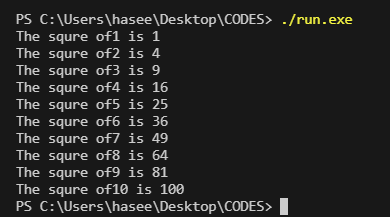


**Program #12:**

**C++ Program to Calculate Square of ten numbers using Functions.**

|  |
| --- |
| #include <iostream>  using namespace std;  int calculation\_of\_square(int num)  {  return num \* num;  }  int main()  {  int num,square;  for (int num = 1; num <= 10; num++)  {  square = calculation\_of\_square(num);  cout << "The squre of"<<num << " is "<< square<<endl;  }      return 0;  } |

Output

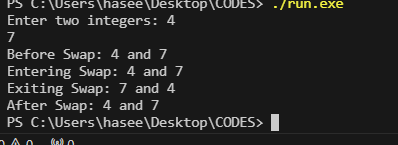


**Program #13:**

**: Write a program that inputs two integers in main ( ) and passes the integers to swap( ) by value. The swap( ) exchanges the values. The main( ) should display the values before and after swapping.**

|  |
| --- |
| #include <iostream>  using namespace std;  void swap\_value(int, int);  int main()  {  int one1,two2;  cout<<"Enter two integers: ";  cin>>one1>>two2;  cout<<"Before Swap: "<<one1<<" and "<<two2<<endl;  swap\_value(one1, two2);  cout<<"After Swap: "<<one1<<" and "<<two2<<endl;  return 0;  }  void swap\_value(int one1, int two2)  {  cout<<"Entering Swap: "<<one1<<" and "<<two2<<endl;  int temp;  temp = one1;  one1 = two2;  two2 = temp;  cout<<"Exiting Swap: "<<one1<<" and "<<two2<<endl;  } |

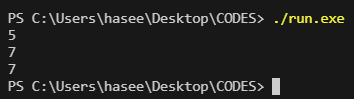
Output



**Program #14:**

|  |
| --- |
| #include <iostream>  using namespace std;  void addition\_of\_two(int &a)  {  a = a+2;  cout << a << endl;  }    int main()  {  int a = 5;  cout << a << endl;  addition\_of\_two(a);  cout << a << endl;  } |

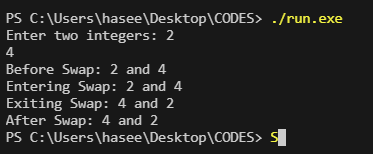
**Output**



**Program #15: Write a program that inputs two integers in main ( ) and passes the integers to swap( ) by reference. The swap( ) exchanges the values. The main( ) should display the values before and after swapping.**

|  |
| --- |
| #include <iostream>  using namespace std;  void swap\_ref(int& value1, int& value2)  {  cout<<"Entering Swap: "<<value1<<" and "<<value2<<endl;  int temp;  temp = value1;  value1 = value2;  value2 = temp;  cout<<"Exiting Swap: "<<value1<<" and "<<value2<<endl;  }  int main()  {  int value1,value2;  cout<<"Enter two integers: ";  cin>>value1>>value2;  cout<<"Before Swap: "<<value1<<" and "<<value2<<endl;  swap\_ref(value1, value2);  cout<<"After Swap: "<<value1<<" and "<<value2<<endl;  return 0;  } |

Ouput



**========================================================================================================================================================================================================================================================================**

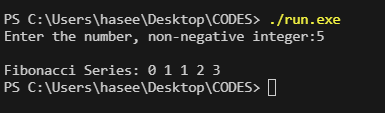
**Program #16:**

**Fibonacci Series Using Recursion in C++**

**Fibonacci number series is the sequence of numbers such that each number is the sum of the two preceding ones starting from zero(0) and one(1).**

|  |
| --- |
| #include <iostream>  using namespace std;  int fibonacci(int i)  {      if (i == 0 || i == 1)      {          return i;      }      else      {          return fibonacci(i - 1) + fibonacci(i - 2);      }  }  int main()  {      int x;      cout << "Enter the number, non-negative integer:";      cin >> x;      while (x < 0)      {          cout << "Please enter a non-negative integer: ";          cin >> x;      }      cout << endl;      cout << "Fibonacci Series: ";      for (int i = 0; i < x; ++i)      {          cout << fibonacci(i) << " ";      }      return 0;  } |

Output



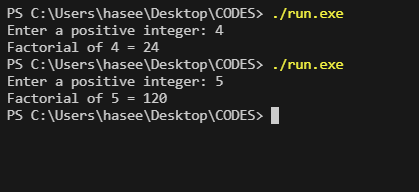
**Program #17:**

**Factorial Program Using Recursion In C++**

**Factorial is the product of an integer and all other integers below it. For example, the factorial of 5 (5!) is equal to 5x4x3x2x1 i.e. 120.**

|  |
| --- |
| #include <iostream>  using namespace std;  int fact(int n);  int main()  {  int n;  cout << "Enter a positive integer: ";  cin >> n;  cout << "Factorial of " << n << " = " << fact(n);  return 0;  }  int fact(int n)  {  if(n > 1)  return n \* fact(n - 1);  else  return 1;  } |

Output



**Program #18:**

**Program To Calculate Number Power Using Recursion In C++**

**In this program, we will calculate the power of the number using the recursion approach where the base and exponent is taken as input by the user.**

|  |
| --- |
| #include <iostream>  #include <cmath>  using namespace std;  int calculate(int, int);  int main()  {  int base, power, result;  cout << "Enter base number: ";  cin >> base;  cout << "Enter power number(positive integer): ";  cin >> power;  result = calculate(base, power);  cout << base << "^" << power << " = " << result;  return 0;  }  int calculate(int base, int power)  {  if (power != 0){  int result = 0;  result += pow(base,power);  return result;  }  else{  return 1;  }  } |

Output

