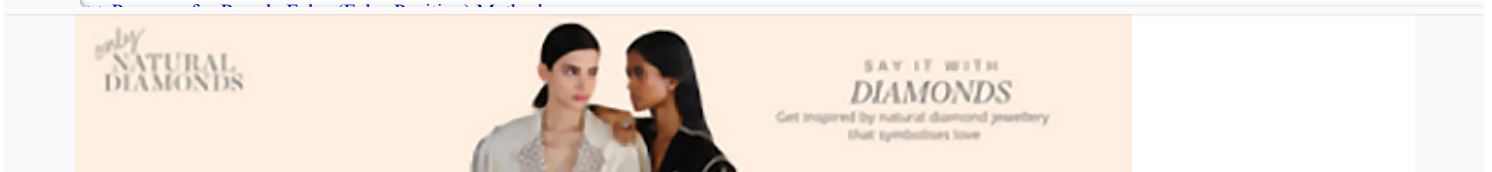
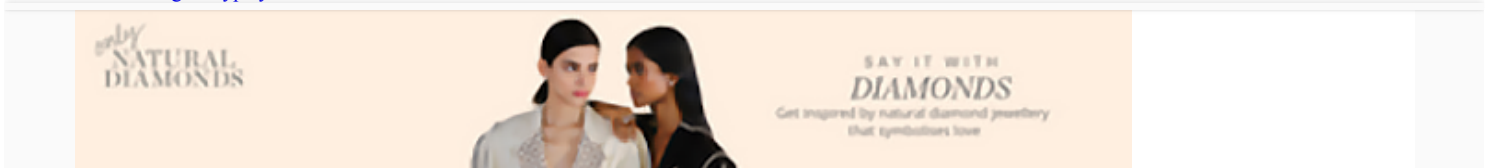


- [Tutorials ▾](#)
  - [Python Programming](#)
  - [C Programming](#)
  - [Numerical Methods](#)
  - [Dart Language](#)
  - [Computer Basics](#)
  - [Flutter](#)
  - [Linux](#)
  - [Deep Learning](#)
- [Examples ▾](#)
  - [C Programming Examples](#)
  - [Python Programming Examples](#)
- [Online Calculator](#)
- [Tutorials ▾](#)
  - [Python Programming](#)
  - [C Programming](#)
  - [Numerical Methods](#)
  - [Dart Language](#)
  - [Computer Basics](#)
  - [Flutter](#)
  - [Linux](#)
  - [Deep Learning](#)
- [Examples ▾](#)
  - [C Programming Examples](#)
  - [Python Programming Examples](#)
- [Online Calculator](#)
- [Algorithm for Bisection Method](#)
- [Pseudocode for Bisection Method](#)
- [C Program for Bisection Method](#)
- [C++ Program for Bisection Method](#)
- [MATLAB Program for Bisection Method](#)
- [Python Program for Bisection Method](#)
- [Bisection Method Advantages](#)
- [Bisection Method Disadvantages](#)
- [Bisection Method Features](#)
- [Convergence of Bisection Method](#)
- [Bisection Method Online Calculator](#)
- [Algorithm for Regula Falsi \(False Position Method\)](#)
- [Pseudocode for Regula Falsi \(False Position\) Method](#)
- [Features of Regula Falsi](#)
- [Falsi Position Advantages](#)
- [False Position Disadvantages](#)
- [Program for Regula False \(False Position\) Method](#)

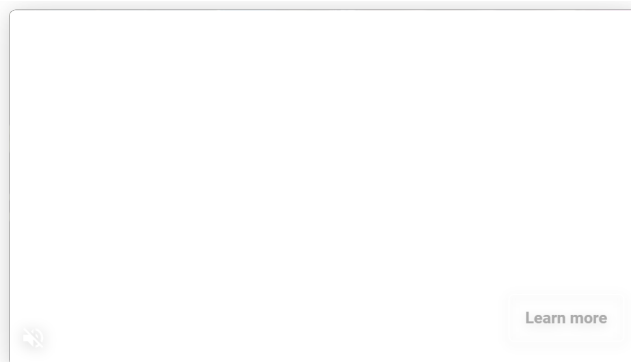


- [Newton Raphson Method C++ Program](#)
- [Newton Raphson Method Python Program](#)
- [Newton-Raphson MATLAB](#)
- [Features of Newton Raphson Method](#)
- [Newton Raphson Advantages](#)
- [Newton Raphson Disadvantages](#)
- [Newton Raphson Method Online Calculator](#)
- [Secant Method Algorithm](#)
- [Secant Method Pseudocode](#)
- [Secant Method C Program](#)
- [Secant Method C++ Program with Output](#)
- [Secant Method Python Program with Output](#)
- [Secant Method Online Calculator](#)
- [Fixed Point Iteration \(Iterative\) Method Algorithm](#)
- [Fixed Point Iteration \(Iterative\) Method Pseudocode](#)
- [Fixed Point Iteration \(Iterative\) Method C Program](#)
- [Fixed Point Iteration \(Iterative\) Python Program](#)
- [Fixed Point Iteration \(Iterative\) Method C++ Program](#)
- [Fixed Point Iteration \(Iterative\) Method Online Calculator](#)
- [Gauss Elimination Method Algorithm](#)
- [Gauss Elimination Method Pseudocode](#)
- [Gauss Elimination C Program](#)
- [Gauss Elimination C++ Program with Output](#)
- [Gauss Elimination Method Python Program with Output](#)
- [Gauss Elimination Method Online Calculator](#)
- [Gauss Jordan Method Algorithm](#)
- [Gauss Jordan Method Pseudocode](#)
- [Gauss Jordan Method C Program](#)
- [Gauss Jordan Method C++ Program](#)
- [Gauss Jordan Method Python Program \(With Output\)](#)
- [Gauss Jordan Method Online Calculator](#)
- [Matrix Inverse Using Gauss Jordan Method Algorithm](#)
- [Matrix Inverse Using Gauss Jordan Method Pseudocode](#)
- [Matrix Inverse Using Gauss Jordan C Program](#)
- [Matrix Inverse Using Gauss Jordan C++ Program](#)
- [Python Program to Inverse Matrix Using Gauss Jordan](#)
- [Matrix Inverse Online Calculator](#)
- [Power Method \(Largest Eigen Value and Vector\) Algorithm](#)
- [Power Method \(Largest Eigen Value and Vector\) Pseudocode](#)
- [Power Method \(Largest Eigen Value and Vector\) C Program](#)
- [Power Method \(Largest Eigen Value and Vector\) C++ Program](#)
- [Power Method \(Largest Eigen Value & Vector\) Python Program](#)
- [Jacobi Iteration Method Algorithm](#)
- [Jacobi Iteration Method C Program](#)
- [Jacobi Iteration Method C++ Program with Output](#)
- [Python Program for Jacobi Iteration](#)
- [Gauss Seidel Iteration Method Algorithm](#)
- [Gauss Seidel Iteration Method C Program](#)
- [Gauss Seidel Iteration Method C++ Program](#)
- [Python Program for Gauss Seidel Iteration Method](#)
- [Python Program for Successive Over Relaxation](#)
- [Forward Difference Table Using C](#)
- [Forward Difference Table Using C++](#)
- [Python Program to Generate Forward Difference Table](#)
- [Python Program to Generate Backward Difference Table](#)
- [Backward Difference Table Using C](#)
- [Backward Difference Table Using C++](#)
- [Lagrange Interpolation Method Algorithm](#)
- [Lagrange Interpolation Method Pseudocode](#)
- [Lagrange Interpolation Method C Program](#)
- [Lagrange Interpolation Method C++ Program](#)
- [Lagrange Interpolation in Python](#)
- [Linear Interpolation Method Algorithm](#)
- [Linear Interpolation Method Pseudocode](#)
- [Linear Interpolation Method C Program](#)
- [Linear Interpolation Method C++ Program with Output](#)
- [Linear Interpolation Method Python Program](#)
- [Linear Regression Method Algorithm](#)
- [Linear Regression Method Pseudocode](#)
- [Linear Regression Method C Program](#)
- [Linear Regression Method C++ Program with Output](#)
- [Linear Regression Python](#)
- [Curve Fitting of Type  \$y=ax^b\$  Algorithm](#)
- [Curve Fitting of Type  \$y=ax^b\$  Pseudocode](#)



- [Curve Fitting  \$y = ab^x\$  C Program](#)
- [Curve Fitting  \$y = ab^x\$  C++ Program](#)
- [Curve Fitting  \$y = ab^x\$  Python Program](#)
- [Derivative Using Forward Difference Formula Algorithm](#)
- [Derivative Using Forward Difference Formula Pseudocode](#)
- [C Program to Find Derivative Using Forward Difference Formula](#)
- [Derivative Using Backward Difference Formula Algorithm](#)
- [Derivative Using Backward Difference Formula Pseudocode](#)
- [C Program to Find Derivative Using Backward Difference Formula](#)
- [Trapezoidal Method for Numerical Integration Algorithm](#)
- [Trapezoidal Method for Numerical Integration Pseudocode](#)
- [Trapezoidal Method C Program](#)
- [Trapezoidal Method C++ Program](#)
- [Trapezoidal Method Python](#)
- [Simpson's 1/3 Rule Algorithm](#)
- [Simpson's 1/3 Rule Pseudocode](#)
- [Simpson's 1/3 Rule C Program](#)
- [Simpson's 1/3 Rule C++ Program](#)
- [Simpson's 1/3 Rule Python](#)
- [Simpson's 3/8 Rule Algorithm](#)
- [Simpson's 3/8 Rule Pseudocode](#)
- [Simpson's 3/8 Rule C Program](#)
- [Simpson's 3/8 Rule Python](#)
- [Simpson's 3/8 Rule C++ Program](#)
- [Euler's Method Algorithm](#)
- [Euler's Method Pseudocode](#)
- [Euler's Method C Program](#)
- [Euler's Method C++ Program](#)
- [Euler's Method Python](#)
- [RK4 Method Python](#)
- [Runge Kutta \(RK\) Algorithm](#)
- [Runge Kutta \(RK\) Pseudocode](#)
- [Runge Kutta \(RK\) C Program](#)
- [Runge Kutta \(RK\) C++ Program](#)

## C Program for Lagrange Interpolation Method (with Output)



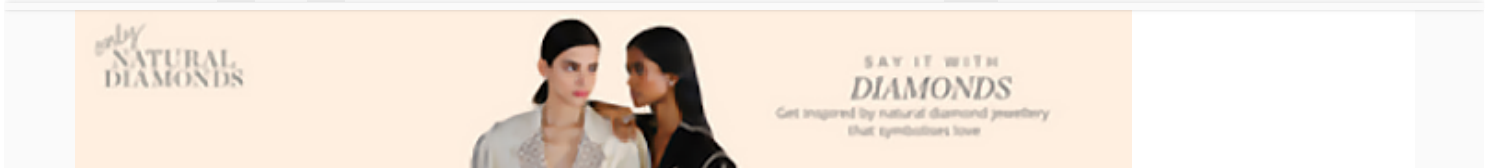
① ×

### Table of Contents

- [Program](#)
- [Output](#)
- [Recommended Readings](#)

**This program implements Lagrange Interpolation Formula in C Programming Language.**

In this C program, **x** and **y** are two array for storing x data and y data respectively. **xp** is interpolation point given by user and



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

void main()
{
    float x[100], y[100], xp, yp=0, p;
    int i,j,n;
    clrscr();
    /* Input Section */
    printf("Enter number of data: ");
    scanf("%d", &n);
    printf("Enter data:\n");
    for(i=1;i<=n;i++)
    {
        printf("x[%d] = ", i);
        scanf("%f", &x[i]);
        printf("y[%d] = ", i);
        scanf("%f", &y[i]);
    }
    printf("Enter interpolation point: ");
    scanf("%f", &xp);
    /* Implementing Lagrange Interpolation */
    for(i=1;i<=n;i++)
    {
        p=1;
        for(j=1;j<=n;j++)
        {
            if(i!=j)
            {
                p = p * (xp - x[j])/(x[i] - x[j]);
            }
        }
        yp = yp + p * y[i];
    }
    printf("Interpolated value at %.3f is %.3f.", xp, yp);
    getch();
}
```

### C Program Output: Lagrange Interpolation

Enter number of data: 5 ↓

Enter data:

$x[1] = 5 \downarrow$

```
y[1] = 150 ↵
```

x[2] = 7 ↵

```
y[2] = 392 ↵
```

```
x[3] = 11 ↓
```

```
y[3] = 1452
```

x[4] = 13 ↓

```
y[4] = 2366
```

x[5] = 17 ↵

$$v_{M51} = 5202 \text{ km/s}$$


## Recommended Readings

1. [Lagrange Interpolation Method Algorithm](#)
2. [Lagrange Interpolation Method Pseudocode](#)
3. [C Program for Lagrange Interpolation Method](#)
4. [C++ Program for Lagrange Interpolation Method](#)
5. [Python Program for Lagrange Interpolation Method](#)

### State-Of-The-Art Technology

MS in Computer Science

Earn A Master's Degree In Computer Science | Take The First Step And Learn More

discover.clarkson.edu

OPEN

- Bisection Method
  - [Algorithm for Bisection Method](#)
  - [Pseudocode for Bisection Method](#)
  - [C Program for Bisection Method](#)
  - [C++ Program for Bisection Method](#)
  - [MATLAB Program for Bisection Method](#)
  - [Python Program for Bisection Method](#)
  - [Bisection Method Advantages](#)
  - [Bisection Method Disadvantages](#)
  - [Bisection Method Features](#)
  - [Convergence of Bisection Method](#)
  - [Bisection Method Online Calculator](#)
- Regula Falsi (False Position) Method
  - [Algorithm for Regula Falsi \(False Position\) Method](#)
  - [Pseudocode for Regula Falsi \(False Position\) Method](#)
  - [Features of Regula Falsi](#)
  - [Falsi Position Advantages](#)
  - [False Position Disadvantages](#)
  - [C Program for Regula False \(False Position\) Method](#)
  - [C++ Program for Regula False \(False Position\) Method](#)
  - [MATLAB Program for Regula False \(False Position\) Method](#)
  - [Python Program for Regula False \(False Position\) Method](#)
  - [Regula Falsi or False Position Method Online Calculator](#)
- Newton Raphson Method
  - [Newton Raphson \(NR\) Method Algorithm](#)
  - [Newton Raphson \(NR\) Method Pseudocode](#)
  - [Newton Raphson Method C Program](#)
  - [Newton Raphson Method C++ Program](#)
  - [Newton Raphson Method Python Program](#)
  - [Newton-Raphson MATLAB](#)
  - [Features of Newton Raphson Method](#)
  - [Newton Raphson Advantages](#)
  - [Newton Raphson Disadvantages](#)
  - [Newton Raphson Method Online Calculator](#)
- Secant Method
  - [Secant Method Algorithm](#)
  - [Secant Method Pseudocode](#)
  - [Secant Method C Program](#)
  - [Secant Method C++ Program with Output](#)
  - [Secant Method Python Program with Output](#)
  - [Secant Method Online Calculator](#)
- Fixed Point Iteration
  - [Fixed Point Iteration \(Iterative\) Method Algorithm](#)

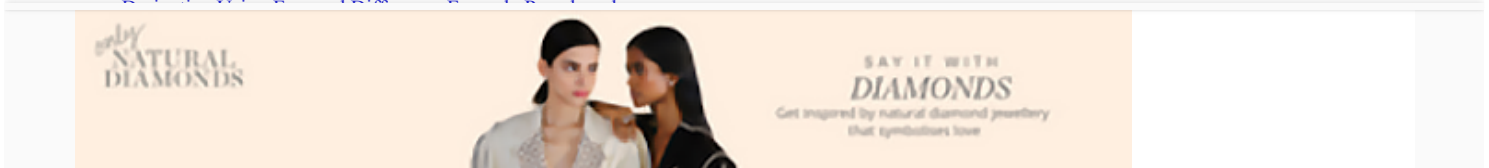
only  
NATURAL  
DIAMONDS



SAY IT WITH  
**DIAMONDS**  
Get inspired by natural diamond jewellery  
that symbolises love

- [Gauss Elimination Method Pseudocode](#)
- [Gauss Elimination C Program](#)
- [Gauss Elimination C++ Program with Output](#)
- [Gauss Elimination Method Python Program with Output](#)
- [Gauss Elimination Method Online Calculator](#)
- Gauss Jordan Method
  - [Gauss Jordan Method Algorithm](#)
  - [Gauss Jordan Method Pseudocode](#)
  - [Gauss Jordan Method C Program](#)
  - [Gauss Jordan Method C++ Program](#)
  - [Gauss Jordan Method Python Program \(With Output\)](#)
  - [Gauss Jordan Method Online Calculator](#)
- Matrix Inverse Using Gauss Jordan
  - [Matrix Inverse Using Gauss Jordan Method Algorithm](#)
  - [Matrix Inverse Using Gauss Jordan Method Pseudocode](#)
  - [Matrix Inverse Using Gauss Jordan C Program](#)
  - [Matrix Inverse Using Gauss Jordan C++ Program](#)
  - [Python Program to Inverse Matrix Using Gauss Jordan](#)
  - [Matrix Inverse Online Calculator](#)
- Power Method
  - [Power Method \(Largest Eigen Value and Vector\) Algorithm](#)
  - [Power Method \(Largest Eigen Value and Vector\) Pseudocode](#)
  - [Power Method \(Largest Eigen Value and Vector\) C Program](#)
  - [Power Method \(Largest Eigen Value and Vector\) C++ Program](#)
  - [Power Method \(Largest Eigen Value & Vector\) Python Program](#)
- Jacobi Iteration Method
  - [Jacobi Iteration Method Algorithm](#)
  - [Jacobi Iteration Method C Program](#)
  - [Jacobi Iteration Method C++ Program with Output](#)
  - [Python Program for Jacobi Iteration](#)
- Gauss Seidel Iteration Method
  - [Gauss Seidel Iteration Method Algorithm](#)
  - [Gauss Seidel Iteration Method C Program](#)
  - [Gauss Seidel Iteration Method C++ Program](#)
  - [Python Program for Gauss Seidel Iteration Method](#)
- Successive Over-Relaxation (SOR)
  - [Python Program for Successive Over Relaxation](#)
- Interpolation
  - [Forward Difference Table Using C](#)
  - [Forward Difference Table Using C++](#)
  - [Python Program to Generate Forward Difference Table](#)
  - [Python Program to Generate Backward Difference Table](#)
  - [Backward Difference Table Using C](#)
  - [Backward Difference Table Using C++](#)
  - [Lagrange Interpolation Method Algorithm](#)
  - [Lagrange Interpolation Method Pseudocode](#)
  - [Lagrange Interpolation Method C Program](#)
  - [Lagrange Interpolation Method C++ Program](#)
  - [Lagrange Interpolation in Python](#)
  - [Linear Interpolation Method Algorithm](#)
  - [Linear Interpolation Method Pseudocode](#)
  - [Linear Interpolation Method C Program](#)
  - [Linear Interpolation Method C++ Program with Output](#)
  - [Linear Interpolation Method Python Program](#)
- Curve Fitting
  - [Linear Regression Method Algorithm](#)
  - [Linear Regression Method Pseudocode](#)
  - [Linear Regression Method C Program](#)
  - [Linear Regression Method C++ Program with Output](#)
  - [Linear Regression Python](#)
  - [Curve Fitting of Type  \$y=ax^b\$  Algorithm](#)
  - [Curve Fitting of Type  \$y=ax^b\$  Pseudocode](#)
  - [Curve Fitting  \$y=ax^b\$  C Program](#)
  - [Curve Fitting  \$y=ax^b\$  Python Program](#)
  - [Curve Fitting  \$y=ax^b\$  C++ Program](#)
  - [Curve Fitting  \$y=ab^x\$  Algorithm](#)
  - [Curve Fitting  \$y=ab^x\$  Pseudocode](#)
  - [Curve Fitting  \$y=ab^x\$  C Program](#)
  - [Curve Fitting  \$y=ab^x\$  C++ Program](#)
  - [Curve Fitting  \$y=ab^x\$  Python Program](#)
- Numerical Differentiation
  - [Derivative Using Forward Difference Formula Algorithm](#)

✓



- [Trapezoidal Method for Numerical Integration Pseudocode](#)
- [Trapezoidal Method C Program](#)
- [Trapezoidal Method C++ Program](#)
- [Trapezoidal Method Python](#)
- [Simpson's 1/3 Rule Algorithm](#)
- [Simpson's 1/3 Rule Pseudocode](#)
- [Simpson's 1/3 Rule C Program](#)
- [Simpson's 1/3 Rule C++ Program](#)
- [Simpson's 1/3 Rule Python](#)
- [Simpson's 3/8 Rule Algorithm](#)
- [Simpson's 3/8 Rule Pseudocode](#)
- [Simpson's 3/8 Rule C Program](#)
- [Simpson's 3/8 Rule Python](#)
- [Simpson's 3/8 Rule C++ Program](#)
- Ordinary Differential Equation
  - [Euler's Method Algorithm](#)
  - [Euler's Method Pseudocode](#)
  - [Euler's Method C Program](#)
  - [Euler's Method C++ Program](#)
  - [Euler's Method Python](#)
  - [RK4 Method Python](#)
  - [Runge Kutta \(RK\) Algorithm](#)
  - [Runge Kutta \(RK\) Pseudocode](#)
  - [Runge Kutta \(RK\) C Program](#)
  - [Runge Kutta \(RK\) C++ Program](#)



## Home and Kitchen Appliances

Bosch One Home Appliances

Bosch

[Shop](#)

### About Us

Codesansar is online platform that provides tutorials and examples on popular programming languages.

### Links

- [About Us](#)
- [Contact Us](#)
- [Privacy Policy](#)
- [Cookies Policy](#)

### Social Media

- [Facebook](#)
- [Twitter](#)
- [Youtube](#)
- [Github](#)
- ✓ [Instagram](#)

