Title: Find the best values of a_0 and a_1 if the straight line $Y = a_0 + a_1x$ fits to the data of (x_i, y_i).

Theoretical Background: (Least Squares Curve Fitting)

Let the set of data points be (x_i, y_i), i = 1, 2, ... m and let the curve given by $y = f(x)$ fits to this data. At $x = x_i$ the given ordinate is y and the corresponding value on the fitting curve is f(x). If e, is the error of approximation at $x = x_i$, then we have $e_i = y_i - f(x_i)$, if we write,

$$S = [y_1 - f(x_1)]^2 + [y_2 - f(x_2)]^2 + \cdots + [y_m - f(x_m)]^2 = e_1^2 + e_2^2 + \cdots + e_m^2$$

Then the method of least squares consists of minimizing S, i.e., the sum of the square of errors. The given values of (x_i, y_i) are $(1, 0.6), (2, 2.4), (3, 3.5), (4, 4.8), (5, 5.7)$.

x_i	y_i	x_i^2	x_iy_i	$(y_i - \bar{y})$	$(y_i - a_0 - a_ix)^2$
1	0.6	1	0.6	7.84	0.0784
2	2.4	4	4.8	1.00	0.0676
3	3.5	9	10.5	0.01	0.0100
4	4.8	16	19.2	1.96	0.0196
5	5.7	25	28.5	5.29	0.0484
15	17.0	55	63.6	16.10	0.2240

From the Table given above we find $\bar{x} = 3,\ \bar{y} = 3.4$.

We know, for least square curve fitting for straight line,

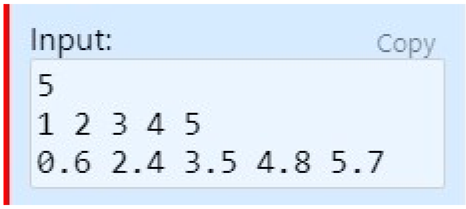
$$a_1 = \frac{m\sum_{i=1}^{m} x_iy_i - \left(\sum_{i=1}^{m} x_i \sum_{i=1}^{m} y_i\right)}{m\sum_{i=1}^{m} x_i^2 - \left(\sum_{i=1}^{m} x_i^2\right)^2} = \frac{5 * 63.6 - 15 * 17}{5 * 55 - 22.5} = 1.26$$

$$a_0 = \bar{y} - a_0\bar{x} = -0.38$$

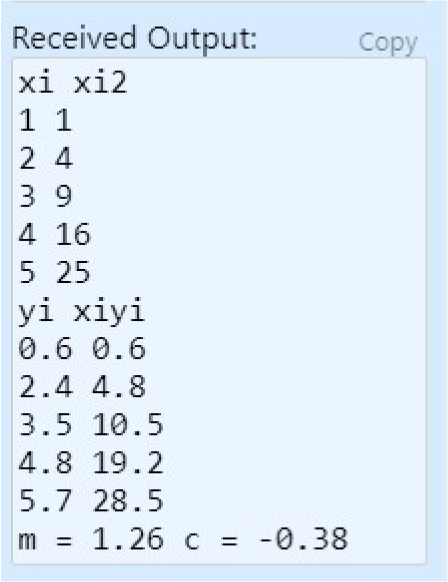
Program:

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    int n;
    cin >> n;
    double x[n], y[n], sumx = 0, sumy = 0;
    double avg1 = 0, avg2 = 0, a = 0, b = 0, c = 0, sq1 = 0;
    cout << "xi "
         << "xi2\n";
    for (int i = 0; i < n; i++)
    {
        cin >> x[i];
        sumx += x[i];
        sq1 += pow(x[i], 2);
        cout << x[i] << " " << pow(x[i], 2) << endl;
    }
    cout << "yi "
         << "xiyi\n";
    for (int i = 0; i < n; i++)
    {
        cin >> y[i];
        sumy += y[i];
        a += (x[i] * y[i]);
        cout << y[i] << " " << (x[i] * y[i]) << endl;
    }
    avg1 = sumx / n;
    avg2 = sumy / n;
    double a1 = ((n * a) - (sumx * sumy)) / ((n * sq1) - pow(sumx, 2));
    double a0 = avg2 - (a1 * avg1);
    cout << "m = " << a1 << " c = " << a0 << endl;
}
```

Input & Output:



```
5
1 2 3 4 5
0.6 2.4 3.5 4.8 5.7
```



```
xi xi2
1 1
2 4
3 9
4 16
5 25
yi xiyi
0.6 0.6
2.4 4.8
3.5 10.5
4.8 19.2
5.7 28.5
m = 1.26 c = -0.38
```

Discussion:

Least squares curve fitting of a straight line involves finding the line those best fits a given set of data points by minimizing the sum of the squared vertical distances between each data point and the line. This technique uses the principle of least squares to determine the optimal slope and intercept of the line, ensuring that it approximates the data as closely as possible. By minimizing the squared errors, this method calculates the parameters that define the straight line, allowing for accurate predictions and a reliable representation of the relationship between variables. Its widespread application stems from its ability to efficiently model linear relationships within datasets across various disciplines, making it a cornerstone of regression analysis and data interpretation.