**PROGRAM 13**

/\*Name:Manjeet Negi Class : G2

Roll no. : 44

\*/

**AIM / OBJECTIVE:** WAP to linear Curve fitting by least – square approximations.

**ALGORITHM:**

1. Start
2. Input the number of data points (n)
3. For Each datapoint I(1to n) :

input x[i]

input y[i]

## Initialize Sums

sumX for the sum of all x-values.

sumX2 for the sum of squares of all x-values. sumY for the sum of all y-values.

sumXY for the sum of products of x[i] and y[i]

## Calculate Required Sums

For each data pointi (1 to n):

Add x[i] to sumX.

Add x[i] \* x[i] to sumX2

Add y[i] to sumY

Add x[i] \* y[i] to sumXY

## Compute a and b

* 1. Calculate the slope b using the formula:

𝑏 = 𝑛 ∗ 𝑠𝑢𝑚𝑋𝑌 − 𝑠𝑢𝑚𝑋 ∗ 𝑠𝑢𝑚𝑌

𝑛 ∗ 𝑠𝑢𝑚𝑋2 − 𝑠𝑢𝑚𝑋 ∗ 𝑠𝑢𝑚𝑋

* 1. Calculate the intercept a using the formula:

𝑎 = 𝑠𝑢𝑚𝑌 − 𝑏 ∗ 𝑠𝑢𝑚𝑋

𝑛

1. Display the values of a (intercept) and b (slope).
2. Display the equation of the best-fit line: 𝒚 = 𝒂 + 𝒃 ∗ 𝒙
3. Stop

## CODE:

## // Fitting y = a + b \* x

## #include <stdio.h>

## #define S 50

## int main()

## {

## int n, i;

## float x[S], y[S], sumX = 0, sumX2 = 0, sumY = 0, sumXY = 0, a, b;

## printf("Enter the number of data points (n): ");

## 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]);

## }

## //Calculating Required Sum

## for (i = 1; i <= n; i++)

## {

## sumX = sumX + x[i];

## sumX2 = sumX2 + x[i] \* x[i];

## sumY = sumY + y[i];

## sumXY = sumXY + x[i] \* y[i];

## }

## // Calculating a and b

## b = (n \* sumXY - sumX \* sumY) / (n \* sumX2 - sumX \* sumX);

## a = (sumY - b \* sumX) / n;

## // Displaying value of a and b

## printf("Values are: a=%0.2f and b = %0.2f", a, b);

## printf("\nEquation of best fit is: y = %0.2f + %0.2fx", a, b);

## return (0);

## }

## OUTPUT:

## 