|  | **MODE FROM STATISTICS IN 10TH CLASS** |
| --- | --- |
| **C**  **O**  **D**  **E** | #include<iostream>  using namespace std;  // Function to calculate and print the mode value Z  void mode()  {  int k=0, h = 0, r = 0, arr1[100], arr2[100], arr3[100], limit;  int f1=0, f2=0, f0=0, l=0;  double dif=0, z=0;  // Input: Get the number of values to be entered  cout << "Enter how many values you want to enter:";  cin >> limit;  cout << "\n\n Example of x = 35-45\n Means 35 is the left side of x and 45 is the right side of x:\n";  // Collect left side of x-values  while (h < limit)  {  r = h + 1;  cout << "\n Enter the left side of x " << r << " value:";  cin >> arr1[h];  h++;  }  h = 0;    cout<<"\n";    // Collect right side of x-values  while (h < limit)  {  r = h + 1;  cout << "\n Enter the right side of x " << r << " value:";  cin >> arr2[h];  h++;  }  h = 0;  cout<<"\n";  // Collect y-values  while (h < limit)  {  r = h + 1;  cout << "\n Enter the value of y " << r << ":";  cin >> arr3[h];  h++;  }  h = 0;    cout<<"\n";  // Display header for the x and y table  cout << "\n HARDIK DHARAIYA 22FOTCA11034 \n\n x \t\t y \n";  cout << "--------------------------\n";  // Print x-values and y-values in two columns  while (h < limit)  {  cout << " " << arr1[h] << "-" << arr2[h] << "\t\t " << arr3[h] << "\n";  h++;  }  // Find the index and value of the maximum y  int max\_y = arr3[0];  int max\_y\_index = 0;  for (k = 1; k < limit; k++)  {  if (arr3[k] > max\_y)  {  max\_y = arr3[k];  max\_y\_index = k;  }  }  // Print the maximum y value and its corresponding x range  cout << "\n Maximum value of y: " << max\_y << endl;  cout << " X range corresponding to max y: " << arr1[max\_y\_index] << "-" << arr2[max\_y\_index] << endl;  // Calculate and print f0, f1, and f2 values  f0 = arr3[max\_y\_index - 1];  cout << " f0 = " << f0 << endl;  f1 = arr3[max\_y\_index];  cout << " f1 = " << f1 << endl;  f2 = arr3[max\_y\_index + 1];  cout << " f2 = " << f2 << endl;  // Calculate values for the model  dif = arr2[0] - arr1[0];  l = arr1[max\_y\_index];  cout << " h = " << dif << endl;  cout << " l = " << l << endl;  int solf1 = f1 \* 2;  int upsideZ = f1 - f0;  int downsideZ = solf1 - f0 - f2;  float solup = (upsideZ \* dif) / downsideZ;  z = l + solup;  // Print the calculated mode value Z  cout << "\n\n Mode(Z) = " << z << endl;  }  // Main function where the program starts  int main()  {  mode(); // Call the mode function  return 0; // Return 0 to indicate successful program execution  } |
| **O**  **U**  **T**  **P**  **T** |  |