```
// Chapter 5 Demo 12
// purpose - computing bowling statistics for a 5 man team which bowls 3 games
//          in a bowling match

#include <iostream>
#include <fstream>
using namespace std;

int main()
{
    int scores[5][3];                       // bowling game scores of all 5 bowlers
                                            //  for all 3 games of series
    int sum = 0;                            // sum of all game's bowled by everyone
                                            //  on the team
    double gameAve = 0.0;                   // game average
    int row = 0;                            // loop variable
    int col = 0;                            // loop variable
    int bowlerNum = 0;                      // number of individual bowler
    int bowlerSum = 0;                      // individual bowler's series score (sum
                                            //  of all 3 games bowled)
    double bowlerAve = 0.0;                 // individual bowler game average
    int gameSum = 0;                        // sum of all 5 bowlers' scores for an
                                            //  individual game
    int gameNum = 0;                        // number of individual game
    int highGame = 0;                       // high game of all bowlers over all games
    double bowlerAverages[5];               // averages of each bowler on team

    for (row = 0; row < 5; row++)           // allowing user to input ALL scores
    {

        for (col = 0; col < 3; col++)
        {
            cout << "Enter next game score: ";
            cin >> scores[row][col];
        }

    }

    for (row = 0; row < 5; row++)           // generating total of ALL  scores
    {
        for (col = 0; col < 3; col++)
        {
            sum = sum + scores[row][col];
        }
    }

    gameAve = sum / 15.0;

    cout << "The individual game average of all 3 bowlers over all 15 games is "
        << gameAve << endl;

    // ****************************************************************

    cout << "Which bowler's average would you like to know? [1, 2, 3, 4, or 5] ";
    cin >> bowlerNum;

    for (col = 0; col < 3; col++)               // computing a single
    {                                           //      bowler's average
        bowlerSum = bowlerSum + scores[bowlerNum - 1][col];
    }

    cout << "His or her average is " << bowlerSum / 3.0 << endl;

    // EXERCISE: Modify the code above so that the program displays which
    //          bowler's (#1, 2, 3, 4, or 5) average is being displayed.

    // ****************************************************************

    cout << "Which game would you like the average of? [1, 2, or 3] ";
    cin >> gameNum;

    gameSum = 0;

    for (row = 0; row < 5; row++)               // computing a game average
    {
        gameSum = gameSum + scores[row][gameNum];
```

```
    }

    cout << "The average of that game is " << gameSum / 5.0 << endl;

    // EXERCISE: Modify the code above so that the program displays which
    //           game's (#1, 2, or 3) average is being displayed.

    // ****************************************************************

    bowlerSum = 0;                              // reinitializing bowlerSum since
                                                //      it was used earlier

    for (row = 0; row < 5; row++)               // computing & displaying each
                                                //      bowler's average and permanently
                                                //      storing these averages in a
    {                                           //      parallel array
        bowlerSum = 0;

        for (col = 0; col < 3; col++)
        {
            bowlerSum = bowlerSum + scores[row][col];
        }

        bowlerAverages[row] = bowlerSum / 3.0;
        cout << "Bowler #" << row + 1 << "'s average is " << bowlerAverages[row] << endl;
    }

    // ****************************************************************

    for (row = 0; row < 5; row++)               // searching for and displaying the
                                                //      highest game bowled
    {

        for (col = 0; col < 3; col++)
        {

            if (scores[row][col] > highGame)
            {
                highGame = scores[row][col];
            }

        }

    }

    cout << "The highest game was " << highGame << endl;

    // EXERCISE: Modify the code above so that the program displays which bowler
    //              bowled the highest game.

    return 0;
}// end of main
```