```
// Filename Fread_Final_Project.java
// Written by Justin Fread
// Written on 7/28/18

import java.util.*;
import java.text.NumberFormat;

public class Main {

  public static void main(String[] args) {

    Scanner userInput = new Scanner(System.in);

    int numOfGrades = 0;
    int numOfStudents = 0;

    // prompt the user to enter the number students and use numOfStudents to
    // set size of ID_average array
    System.out.print("Please enter the number of students in this course: ");
    numOfStudents = userInput.nextInt();

    // continue accepting number of students for new classes until user enters -1
    while(numOfStudents != -1) {

      String[][] ID_average = new String[numOfStudents][2];

      System.out.print("How many grades will be entered " +
                  "for each student? ");
      numOfGrades = userInput.nextInt();
      userInput.nextLine();

      // populate ID_average array with student ID numbers and corresponding
      // student averages
      for(int row = 0; row < ID_average.length; row++) {

        System.out.print("\nStudent ID: ");

        for(int column = 0; column < ID_average[row].length; column++) {

          ID_average[row][column] = userInput.nextLine();

          if(column < ID_average[row].length - 1) {
            column++;
            ID_average[row][column] = calcuLateAverage(numOfGrades);

            System.out.print(ID_average[row][column - 1] + "       " +
                             ID_average[row][column] + "\n");


          }

        }

      }

      System.out.println();

      // calculate and display class information
      calcClassAverage(ID_average, numOfStudents);
      classMedian(ID_average, numOfStudents);
      calcScholarship(ID_average);

      // prompt the user for the number of students for the next class or a
      // -1 to exit the program
      System.out.print("\nPlease enter the number of students for the next " +
              "course or a -1 to quit: ");
      numOfStudents = userInput.nextInt();
      userInput.nextLine();

    }

  }

  // calculateAverage calls calcLetterGrade
  public static String calculateAverage(int numOfGrades) {

    Scanner input = new Scanner(System.in);
    double grades, total, average;
    total = 0;

    for(int i = 0; i < numOfGrades; i++) {
      System.out.print("Grade " + (i + 1) + ":    ");
      grades = input.nextDouble();
      if(grades < 0 || grades > 100) {
        System.out.println("Grades must be from 0 - 100");
        i--;
      }
      else {
        total += grades;
      }
    }

    average = (double)total / numOfGrades;

    calcLetterGrade(average);

    return Double.toString(average);

  }

  public static void calcLetterGrade(double average) {

    char letterGrade;

    if(average >= 90) {
      letterGrade = 'A';
    }
    else if(average >= 80) {
      letterGrade = 'B';
    }
    else if(average >= 70) {
      letterGrade = 'C';
    }
    else if(average >= 60) {
      letterGrade = 'D';
    }
    else {
      letterGrade = 'F';
    }

    System.out.println("\nLetterGrade    ID Number    Average"
            + "\n----------------------------------");
    System.out.print("      " + letterGrade + "\t\t");

  }
  // calcClassAverage calls highestLowest
  public static void calcClassAverage(String[][] ID_average, int numOfStudents) {

    double classAverage, total = 0;

    for(int row = 0; row < ID_average.length; row++) {
      for(int column = 1; column < ID_average[row].length; column += 2) {

        total += Double.parseDouble(ID_average[row][column]);

      }
    }

    classAverage = (double)total / numOfStudents;
    System.out.println("Class Average: " + classAverage);

    highestLowestAvg(ID_average, classAverage);

  }

  public static void highestLowestAvg(String[][] ID_average, double classAvg) {

    double highest = Double.parseDouble(ID_average[0][1]);
    double lowest = Double.parseDouble(ID_average[0][1]);
    int aboveAvg = 0;
    int aboveC = 0;

    for(int row = 0; row < ID_average.length; row++) {
      for(int column = 1; column < ID_average[row].length; column += 2) {

        if(Double.parseDouble(ID_average[row][column]) > highest) {
          highest = Double.parseDouble(ID_average[row][column]);
        }
        if(Double.parseDouble(ID_average[row][column]) < lowest) {
          lowest = Double.parseDouble(ID_average[row][column]);
        }
        if(Double.parseDouble(ID_average[row][column]) >= classAvg) {
          aboveAvg++;
        }
        if(Double.parseDouble(ID_average[row][column]) >= 70) {
          aboveC++;
        }

      }
    }

    System.out.println("Students Above Average: " + aboveAvg +
                    "\nStuents Avove a C: " + aboveC);
    System.out.println("Highest Average: " + highest + "\nLowest Average:  " +
                       lowest);

  }

  public static void classMedian(String[][] ID_average, int numOfStudents) {

    double[] averages = new double[numOfStudents];
    int index = 0;
    int mean;

    for(int row = 0; row < ID_average.length; row++) {
      for(int column = 1; column < ID_average[row].length; column += 2) {

        averages[index] = Double.parseDouble(ID_average[row][column]);
        index++;

      }
    }

    Arrays.sort(averages);
    mean = numOfStudents / 2;
    System.out.println("The Class Mean is: " + averages[mean]);

  }

  public static void calcScholarship(String[][] ID_average) {

    NumberFormat curFmt = NumberFormat.getCurrencyInstance();
    double scholarship;

    for(int row = 0; row < ID_average.length; row++) {
       for(int column = 1; column < ID_average[row].length; column += 2) {

        if(Double.parseDouble(ID_average[row][column]) > 82.5) {
           scholarship = Double.parseDouble(ID_average[row][column]) * 3.25;
           System.out.println(curFmt.format(scholarship) +
           " Scholarship Awwarded to ID# " +
           ID_average[row][column - 1]);
        }

      }

    }

  }

}
```