```
/*
 * hw4.c
 *
 *  Created on: Oct 4, 2014
 *      Author: Scot Matson
 *      Assign: 4
 *      Course: cs49c
 *     Section: 1
 *        Desc: Program for calculating statistics
related to student
 *        assignment scores.
 */

#define NROW 100
#define NCOL 20

int i, j;
int hws[NROW][NCOL];
int totals[NROW];
double averages[NCOL];

/**
 * Calculates the highest assignment score and retu
rns
 * the student ID#
 */
int topscore() {
    int topStudent = 0;
    int highScore = 0;
    setTotals();

    for (i = (NROW - 1); i >= 0; --i) {
        if(totals[i] >= highScore) {
            highScore = totals[i];
            topStudent = i;
        }
    }
    return topStudent;
}

/**
 * Calculates the lowest average assignment score
```

```
 */
int toughesthomework() {
    double lowestAvg = 0;
    double lowScore = 100.00;
    setAverages();

    for (i = (NCOL - 1); i >= 0; --i) {
        if (averages[i] < lowScore) {
            lowScore = averages[i];
            lowestAvg = i;
        }
    }
    return lowestAvg;
}

/**
 * Returns the number of students who have achieved
 the highest
 * assignment scores.
 */
int numhighest() {
    int numTopScores = 0;
    int highestScore = 0;
    for (i = 0; i < NROW; ++i) {
        for (j = 0; j < NCOL; ++j) {
            if (hws[i][j] > highestScore) {
                highestScore = hws[i][j];
                numTopScores = 1;
            }
            else if (hws[i][j] == highestScore) {
                ++numTopScores;
            }
        }
    }

    return numTopScores;
}

/**
 * Populates totals[n] with each students highest s
core
 */
```

```
void setTotals() {
    int studentTopScore;

    for (i = 0; i < NROW; ++i) {
        studentTopScore = 0;
        for (j = 0; j < NCOL; ++j) {
            studentTopScore += hws[i][j];
        }
        totals[i] = studentTopScore;
    }
}

/**
 * Populates averages[n] with the total average of
each assignment
 */
void setAverages() {
    double assnSum;

    for (j = 0; j < NCOL; ++j) {
        assnSum = 0;
        for (i = 0; i < NROW; ++i) {
            assnSum += hws[i][j];
        }
        averages[j] = (assnSum / NROW);
    }
}
```