

```

/* BY SUBMITTING THIS FILE TO CARMEN, I CERTIFY THAT I HAVE PERFORMED ALL OF
THE WORK TO CREATE THIS FILE AND/OR DETERMINE THE ANSWERS FOUND WITHIN
THIS FILE MYSELF WITH NO ASSISTANCE FROM ANY PERSON (OTHER THAN THE
INSTRUCTOR OR GRADERS OF THIS COURSE) AND I HAVE STRICTLY ADHERED TO THE
TENURES OF THE OHIO STATE UNIVERSITY'S ACADEMIC INTEGRITY POLICY.
*/
#include "lab4.h"
Node* read_student_data(char *name, int id, float a1, float a2, float a3, float b1,
float b2, float b3, float c1, float c2, float c3, float d1, float d2, float d3) {
    struct Data status; /*The structure with each student's data*/
    int count; /*Number of scores in each category for each student*/
    /*Temporary variables to store the cumulative for each category*/
    float cumulative;
    float a, b, c, d;
    int i; /*Loop control variable*/
    Node* current;
    i = 0;
    /*Reads in the characters in the student name and populates it into the
array*/
    while(name[i] != '\0') {
        if (name[i] != '\0') {
            status.student_name[i] = name[i];
            i++;
        }
    }
    status.student_name[i] = '\0';
    /*Initialize all other members of the structre*/
    status.student_ID = id;
    status.cat1.score1 = a1;
    status.cat1.score2 = a2;
    status.cat1.score3 = a3;
    status.cat2.score1 = b1;
    status.cat2.score2 = b2;
    status.cat2.score3 = b3;
    status.cat3.score1 = c1;
    status.cat3.score2 = c2;
    status.cat3.score3 = c3;
    status.cat4.score1 = d1;
    status.cat4.score2 = d2;
    status.cat4.score3 = d3;
    /*Calculates the cumulative score for each category omitting any score listed
as -1*/
    /*Category 1*/
    cumulative = 0;
    count = 0;
    if (a1 != -1) {
        cumulative += a1;
        count++;
    }
    if (a2 != -1) {
        cumulative += a2;
        count++;
    }
    if (a3 != -1) {
        cumulative += a3;
        count++;
    }
    if (count != 0) {
        cumulative /= count;
    }
    else {
        cumulative = -1;
    }
}

```

```
}
status.cat1.cumulative = cumulative;
/*Category 2*/
cumulative = 0;
count = 0;
if (b1 != -1) {
    cumulative += b1;
    count++;
}
if (b2 != -1) {
    cumulative += b2;
    count++;
}
if (b3 != -1) {
    cumulative += b3;
    count++;
}
if (count != 0) {
    cumulative /= count;
}
else {
    cumulative = -1;
}
status.cat2.cumulative = cumulative;
/*Category 3*/
cumulative = 0;
count = 0;
if (c1 != -1) {
    cumulative += c1;
    count++;
}
if (c2 != -1) {
    cumulative += c2;
    count++;
}
if (c3 != -1) {
    cumulative += c3;
    count++;
}
if (count != 0) {
    cumulative /= count;
}
else {
    cumulative = -1;
}
status.cat3.cumulative = cumulative;
/*Category 4*/
cumulative = 0;
count = 0;
if (d1 != -1) {
    cumulative += d1;
    count++;
}
if (d2 != -1) {
    cumulative += d2;
    count++;
}
if (d3 != -1) {
    cumulative += d3;
    count++;
}
if (count != 0) {
```

```
        cumulative /= count;
    }
    else {
        cumulative = -1;
    }
    status.cat4.cumulative = cumulative;
    /*Calculates the current grade based on the following weighing system.
    Category 1: 15%
    Category 2: 30%
    Category 3: 20%
    Category 4: 35%
    */
    a = status.cat1.cumulative;
    b = status.cat2.cumulative;
    c = status.cat3.cumulative;
    d = status.cat4.cumulative;
    if (a == -1) {
        a = 100;
    }
    if (b == -1) {
        b = 100;
    }
    if (c == -1) {
        c = 100;
    }
    if (d == -1) {
        d = 100;
    }
    status.current_grade = 0.15 * a + 0.3 * b + 0.2 * c + 0.35 * d;
    /*All final grades are set to -1 when reading in data*/
    status.final_grade = -1;
    /*Allocates a node for the current student data*/
    current = allocate_node(status);
    return current;
}
```