```
/* 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 ASSISSTANCE 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"
/*Recalculates all sudents' grades*/
void option5(Node *head, char *categories) {
        Node *traversePtr;
        int count; /*Number of scores for each student in each category*/
        float cumulative; /*Temporary variable to store the cumulative score for each
category for each student*/
        float grade; /*Temporary variable to store the student's grade*/
        /*Traverses through the linked list*/
        traversePtr = head;
        while (traversePtr != NULL) {
                count = 0;
                cumulative = 0;
                /*Prints out each student's name*/
                printf("Student Name: %-23s\t", traversePtr->student.student_name);
                /*Recalculates the cumulative score for each category omitting scores
listed as -1. If all individual scores are -1, the category cumulative score is -1.*/
                /*Category 1*/
                if (traversePtr->student.cat1.score1 != -1) {
                        cumulative += traversePtr->student.cat1.score1;
                        count++;
                }
                if (traversePtr->student.cat1.score2 != -1) {
                        cumulative += traversePtr->student.cat1.score2;
                        count++;
                }
                if (traversePtr->student.cat1.score3 != -1) {
                        cumulative += traversePtr->student.cat1.score3;
                        count++;
                }
                if (count != 0) {
                        cumulative /= count;
                }
                else {
                        cumulative = -1;
                }
                traversePtr->student.cat1.cumulative = cumulative;
                /*After recalculating each category, prints out the cumulative for
that category. Repeats this step for the other 3 categories*/
                printf("%s Cumulative: %-8.2f\t", (char*)categories, traversePtr-
>student.cat1.cumulative);
                /*Category 2*/
                count = 0;
                cumulative = 0;
                if (traversePtr->student.cat2.score1 != -1) {
                        cumulative += traversePtr->student.cat2.score1;
                        count++;
                }
                if (traversePtr->student.cat2.score2 != -1) {
                        cumulative += traversePtr->student.cat2.score2;
                        count++;
                }
                if (traversePtr->student.cat2.score3 != -1) {
                        cumulative += traversePtr->student.cat2.score3;
                        count++;
                }
                if (count != 0) {
                        cumulative /= count;
                }
                else {
                        cumulative = -1;
                }
                traversePtr->student.cat2.cumulative = cumulative;
                printf("%s Cumulative: %.2f\t", (char*)categories+15, traversePtr-
>student.cat2.cumulative);
                /*Category 3*/
                count = 0;
                cumulative = 0;
                if (traversePtr->student.cat3.score1 != -1) {
                        cumulative += traversePtr->student.cat3.score1;
                        count++;
                }
                if (traversePtr->student.cat3.score2 != -1) {
                        cumulative += traversePtr->student.cat3.score2;
                        count++;
                }
                if (traversePtr->student.cat3.score3 != -1) {
                        cumulative += traversePtr->student.cat3.score3;
                        count++;
                }
                if (count != 0) {
                        cumulative /= count;
                }
                else {
                        cumulative = -1;
                }
                traversePtr->student.cat3.cumulative = cumulative;
                printf("%s Cumulative: %-8.2f\t", (char*)categories + 30, traversePtr-
>student.cat3.cumulative);
                /*Category 4*/
                count = 0;
                cumulative = 0;
                if (traversePtr->student.cat4.score1 != -1) {
                        cumulative += traversePtr->student.cat4.score1;
                        count++;
                }
                if (traversePtr->student.cat4.score2 != -1) {
                        cumulative += traversePtr->student.cat4.score2;
                        count++;
                }
                if (traversePtr->student.cat4.score3 != -1) {
                        cumulative += traversePtr->student.cat4.score3;
                        count++;
                }
                if (count != 0) {
                        cumulative /= count;
                }
                else {
                        cumulative = -1;
                }
                traversePtr->student.cat4.cumulative = cumulative;
                printf("%s Cumulative: %-8.2f", (char*)categories + 45, traversePtr-
>student.cat4.cumulative);
                /*Recalculates the current overall grade. If any cumulative is -1,
then it becomes 100*/
                grade = 0;
                if (traversePtr->student.cat1.cumulative != -1) {
                        grade += 0.15 * traversePtr->student.cat1.cumulative;
                }
                else {
                        grade += 15;
                }
                if (traversePtr->student.cat2.cumulative != -1) {
                        grade += 0.3 * traversePtr->student.cat2.cumulative;
                }
                else {
                        grade += 30;
                }
                if (traversePtr->student.cat3.cumulative != -1) {
                        grade += 0.2 * traversePtr->student.cat3.cumulative;
                }
                else {
                        grade += 20;
                }
                if (traversePtr->student.cat4.cumulative != -1) {
                        grade += 0.35 * traversePtr->student.cat4.cumulative;
                }
                else {
                        grade += 35;
                }
                traversePtr->student.current_grade = grade;
                /*Prints out the current overall grade*/
                printf("Current Grade is: %-8.2f\n", traversePtr-
>student.current_grade);
                traversePtr->student.final_grade = -1;
                traversePtr = traversePtr->next;
        }
}
```