```c
/* 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"
/*Prints all student records sorted by ID number*/
void option3(Node* head, char *categories) {
        Node *traversePtr;
        float totals[4]; /*Array to store the totals for each category*/
        int counts[4]; /*Array to store the number of students for each category*/
        float averages[4]; /*Array to store the average scores for each category*/
        int num; /*Array to count number of students when storing the average overall
grade*/
        int i; /*Loop control variable*/
        float grade; /*Sum of all student grades*/
        printHeader(categories);
        traversePtr = head;
        /*Initialize the totals and counts arrays*/
        for (i = 0; i < 4; i++) {
                counts[i] = 0;
                totals[i] = 0;
        }
        grade = 0;
        num = 0;
        /*Traverses through the list*/
        while (traversePtr != NULL) {
                printStudent(traversePtr);
                printf("\n");
                /*Creates the average score fore each category across all students.
If no score, then it is a zero.*/
                if (traversePtr->student.cat1.cumulative != -1) {
                        totals[0] += traversePtr->student.cat1.cumulative;
                }
                counts[0]++;
                if (traversePtr->student.cat2.cumulative != -1) {
                        totals[1] += traversePtr->student.cat2.cumulative;
                }
                counts[1]++;
                if (traversePtr->student.cat3.cumulative != -1) {
                        totals[2] += traversePtr->student.cat3.cumulative;
                }
                counts[2]++;
                if (traversePtr->student.cat4.cumulative != -1) {
                        totals[3] += traversePtr->student.cat4.cumulative;
                }
                counts[3]++;
                /*Calculates average overall grade across all students*/
                grade += traversePtr->student.current_grade;
                num++;
                traversePtr = traversePtr->next;
        }
        /*Calculates the average for each category*/
        for (i = 0; i < 4; i++) {
                averages[i] = totals[i]/counts[i];
        }
        grade /= num;
        /*Prints the summary line*/
        printf("\nClass averages for %s: %.2f, %s: %.2f, %s: %.2f, %s: %.2f Current
Grade: %.2f\n", (char*)categories, averages[0], (char*)categories+15, averages[1],
(char*)categories+30, averages[2], (char*)categories+45, averages[3], grade);
}
```