C Programming

UID: 24MCI10204

Name: Rahul Saxena

Branch: 24MCA – AI & ML

Question 1: Create a program to manage student records.

It should:

- Use a struct to store: roll number, name, marks (array of 3 subjects), and average.
- Dynamically allocate memory for N students using malloc() Calculate average marks.
- Store all student data into a file(students.txt) using printf() then, read and display data from the file using scanf()

Key concepts used:

Structure, array, dynamic memory allocation file handling string handing.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define SUBJECTS 3
#define NAME_LEN 100
struct Student {
   int roll;
   char name[NAME_LEN];
   float marks[SUBJECTS];
   float average;
};

int main() {
   int n;
   printf("Enter number of students: ");
   scanf("%d", &n);

   struct Student* students = (struct Student*) malloc(n * sizeof(struct Student));

   if (students == NULL) {
      printf("Memory allocation failed.\n");
      return 1;
   }
   for (int i = 0; i < n; i++) {
      printf("\n--- Enter details for student %d ---\n", i + 1);
      printf("Roll number: ");
      scanf("%d", &students[i].roll);
      printf("Name: ");
      getchar();
      fgets(students[i].name, NAME_LEN, stdin);
      students[i].name[strcspn(students[i].name, "\n")] = '\0';
      float total = 0;
      for (int j = 0; j < SUBJECTS; j++) {
         printf("Marks for subject %d: ", j + 1);
         scanf("%f", &students[i].marks[j]);
         total += students[i].marks[j];
```

```
        }
        students[i].average = total / SUBJECTS;
    }
    FILE* fptr = fopen("students.txt", "w");
    if (fptr == NULL) {
        printf("File could not be opened for writing.\n");
        free(students);
        return 1;
    }

    for (int i = 0; i < n; i++) {
        fprintf(fptr, "%d %s %.2f %.2f %.2f %.2f\n",
            students[i].roll,
            students[i].name,
            students[i].marks[0],
            students[i].marks[1],
            students[i].marks[2],
            students[i].average
        );
    }
    fclose(fptr);
    free(students);
    printf("\n--- Student Records from File ---\n");

    struct Student temp;
    FILE* rptr = fopen("students.txt", "r");
    if (rptr == NULL) {
        printf("File could not be opened for reading.\n");
        return 1;
    }

    while (fscanf(rptr, "%d %s %f %f %f %f",
            &temp.roll,
            temp.name,
            &temp.marks[0],
            &temp.marks[1],
            &temp.marks[2],
            &temp.average) == 6) {

        printf("\nRoll: %d\nName: %s\nMarks: %.2f, %.2f, %.2f\nAverage: %.2f\n",
            temp.roll,
            temp.name,
            temp.marks[0],
            temp.marks[1],
            temp.marks[2],
            temp.average
        );
    }

    fclose(rptr);
    return 0;
}
```

Question 2: Move zeros to the end of the array Given n array move all zero elements to the end without changing the relative order of non zero elements.

Code:

```
#include <stdio.h>

void moveZerosToEnd(int arr[], int n) {
   int index = 0;
   for (int i = 0; i < n; i++) {
      if (arr[i] != 0) {
         arr[index++] = arr[i];
      }
   }
   while (index < n) {
      arr[index++] = 0;
   }
}

int main() {
   int arr[100], n;

   printf("Enter size of array: ");
   scanf("%d", &n);

   printf("Enter %d elements:\n", n);
   for (int i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
   }

   moveZerosToEnd(arr, n);

   printf("Array after moving zeros to the end:\n");
   for (int i = 0; i < n; i++) {
      printf("%d ", arr[i]);
   }

   return 0;
}
```