

Русенски университет „Ангел Кънчев“  
Факултет „Електротехника, електроника и автоматика“  
Дисциплина „Програмиране“

**Курсова задача 2 - Програмиране**

Име: Константин Христианов  
Ганев  
Фак. Номер: 243020  
Група: 26А

Курс 1

Проверил: .....

/гл. ас. д-р инж. Сергей Антонов/

**2024.12.15**

## Задание

№ 13. За група студенти са зададени име, ЕГН, фак. №, оценките по 10 дисциплини, среден успех. Да се създаде програма, която избира от меню и използва функции за:

- създаване на файл с информация за групата студенти, без да се въвежда ср. успех;
- изчисляване на средния успех на всеки един от студентите;
- извеждане на фак. № и средния успех на всички студентки (дами).

# Изисквания към програмата

## I. Желан изход

Програмата трябва да предоставя следния изход:

### 1. Създаване на файл с информация за студентите:

- Данните за всички студенти (име, ЕГН, факултетен номер, оценки) се записват във файл students.txt във формат:

```
Име ЕГН фак. № оценка1 оценка2 ... оценка10
```

### 2. Изчисляване на средния успех

- Средният успех на всеки студент се изчислява и извежда на екрана във формат:

```
Name: <име>, Average Grade: <среден успех>
```

### 3. Извеждане на студентките (дами):

- Програмата извежда факултетния номер и средния успех на всички студентки в групата в следния формат:

```
Name: <име>, Faculty Number: <фак. №>, Average Grade: <среден  
успех>
```

- Ако няма студентки, изходът ще бъде:

```
No female students found.
```

## II. Необходими входни данни

Програмата изисква следните входни данни:

### 1. От потребителя (в реално време, чрез конзолата):

- Брой студенти
- Име, ЕГН, факултетен номер и оценки (10 на брой) за всеки студент.

### 2. Формат на данните:

- Име:** Текстов низ до 50 символа.
- ЕГН:** Текстов низ с точно 10 цифри.
- Факултетен номер:** Цяло число.
- Оценки:** 10 числа с плаваща запетая, представляващи оценките на студента.

## III. Необходима обработка

Обработката в програмата включва:

### 1. Създаване на файл с информация за студентите:

- Четене на входни данни от потребителя и запис във файл.
- Запис на данните в указан формат без изчисляване на среден успех.

## **2. Изчисляване на средния успех:**

- Обхождане на оценките на всеки студент и изчисляване на средноаритметичната стойност.

## **3. Условни оператори:**

- Определяне на пола на студентите по деветия знак на ЕГН.
  - Ако деветият знак е нечетно число, студентът е жена.
- Проверка дали са налични студентки.

## **4. Видове използвани цикли:**

- **For** цикли за обхождане на оценките на студентите и изчисляване на среден успех.
- **For** цикли за обхождане на списъка със студенти при търсене на студентки.
- **Do-While** цикъл за реализация на основното меню на програмата.

## **5. Сортиране (не е включено, но може да бъде добавено):**

- Ако е необходимо, данните могат да бъдат сортирани по среден успех или друг критерий.

## **6. Потребителски интерфейс:**

- Реализиран чрез текстово меню с опции:
  - 1. Създаване на файл.
  - 2. Изчисляване и извеждане на среден успех.
  - 3. Извеждане на студентките.
  - 4. Изход от програмата.

## Текст на програмата

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

const int MAX_STUDENTS = 100;
const int DISCIPLINES = 10;

struct Student {
    char name[50];
    char EGN[11];
    int facNumber;
    float grades[DISCIPLINES];
    float averageGrade;
};

void createStudentFile(Student students[], int* count) {
    printf("Enter number of students: ");
    scanf_s("%d", count);

    FILE* file;
    fopen_s(&file, "students.txt", "w");
    if (!file) {
        printf("Error creating file!\n");
        return;
    }

    for (int i = 0; i < *count; i++) {
        printf("Enter name for student %d: ", i + 1);
        scanf_s("                %[^\\n]s",          students[i].name,
(unsigned)sizeof(students[i].name));

        printf("Enter EGN for student %d: ", i + 1);
        scanf_s("%s",                                students[i].EGN,
(unsigned)sizeof(students[i].EGN));

        printf("Enter faculty number for student %d: ", i + 1);
        scanf_s("%d", &students[i].facNumber);

        printf("Enter grades for %d disciplines for student %d:\\n",
DISCIPLINES, i + 1);
        for (int j = 0; j < DISCIPLINES; j++) {
            printf("Grade %d: ", j + 1);
            scanf_s("%f", &students[i].grades[j]);
        }

        fprintf(file, "%s %s %d ", students[i].name, students[i].EGN,
students[i].facNumber);
        for (int j = 0; j < DISCIPLINES; j++) {
            fprintf(file, "%.2f ", students[i].grades[j]);
        }
    }
}
```

```

    }
    fprintf(file, "\n");
}
fclose(file);
printf("File 'students.txt' created successfully.\n");
}

void calculateAverageGrade(Student students[], int count) {
    for (int i = 0; i < count; i++) {
        float sum = 0;
        for (int j = 0; j < DISCIPLINES; j++) {
            sum += students[i].grades[j];
        }
        students[i].averageGrade = sum / DISCIPLINES;
    }

    printf("\nAverage grades for all students:\n");
    for (int i = 0; i < count; i++) {
        printf("Name: %s, Average Grade: %.2f\n", students[i].name,
students[i].averageGrade);
    }
}

void displayFemaleStudents(Student students[], int count) {
    printf("\nFemale students:\n");
    bool found = false;

    for (int i = 0; i < count; i++) {
        int genderDigit = students[i].EGN[8] - '0';

        if (genderDigit % 2 != 0) {
            printf("Name: %s, Faculty Number: %d, Average Grade: %.2f\n",
students[i].name, students[i].facNumber,
students[i].averageGrade);
            found = true;
        }
    }

    if (!found) {
        printf("No female students found.\n");
    }
}

// Основната програма
void main() {
    Student students[MAX_STUDENTS];
    int count = 0;
    int choice;

    do {
        printf("\nMenu:\n");

```

```

printf("1. Create student file\n");
printf("2. Calculate and display average grades\n");
printf("3. Display women students\n");
printf("4. Exit\n");
printf("Enter your choice: ");
scanf_s("%d", &choice);

switch (choice) {
case 1:
    createStudentFile(students, &count);
    break;
case 2:
    if (count > 0) {
        calculateAverageGrade(students, count);
    }
    else {
        printf("No students available. Please create a student
file first.\n");
    }
    break;
case 3:
    if (count > 0) {
        calculateAverageGrade(students, count);
        displayFemaleStudents(students, count);
    }
    else {
        printf("No students available. Please create a student
file first.\n");
    }
    break;
case 4:
    printf("Exiting program.\n");
    break;
default:
    printf("Invalid choice. Please try again.\n");
}
} while (choice != 4);
}

```

# Тестване

## 1. Create student file

```
Select D:\Софтуър\Mini-Projects\CourseWork2_Year1_Semester1\CourseWork2_Year1_Semester1\Debug\CourseWork2_Year1_Semester1.exe
Menu:
1. Create student file
2. Calculate and display average grades
3. Display women students
4. Exit
Enter your choice: 1
Enter number of students: 2
Enter name for student 1: Konstantin
Enter EGN for student 1: 0645255324
Enter faculty number for student 1: 243020
Enter grades for 10 disciplines for student 1:
Grade 1: 5
Grade 2: 6
Grade 3: 4
Grade 4: 5
Grade 5: 4
Grade 6: 5
Grade 7: 5
Grade 8: 6
Grade 9: 5
Grade 10: 5
Enter name for student 2: Martina
Enter EGN for student 2: 0935426533
Enter faculty number for student 2: 243023
Enter grades for 10 disciplines for student 2:
Grade 1: 5
Grade 2: 5
Grade 3: 4
Grade 4: 3
Grade 5: 6
Grade 6: 6
Grade 7: 4
Grade 8: 5
Grade 9: 5
Grade 10: 5
File 'students.txt' created successfully.
```

## 2. Calculate and display average grades

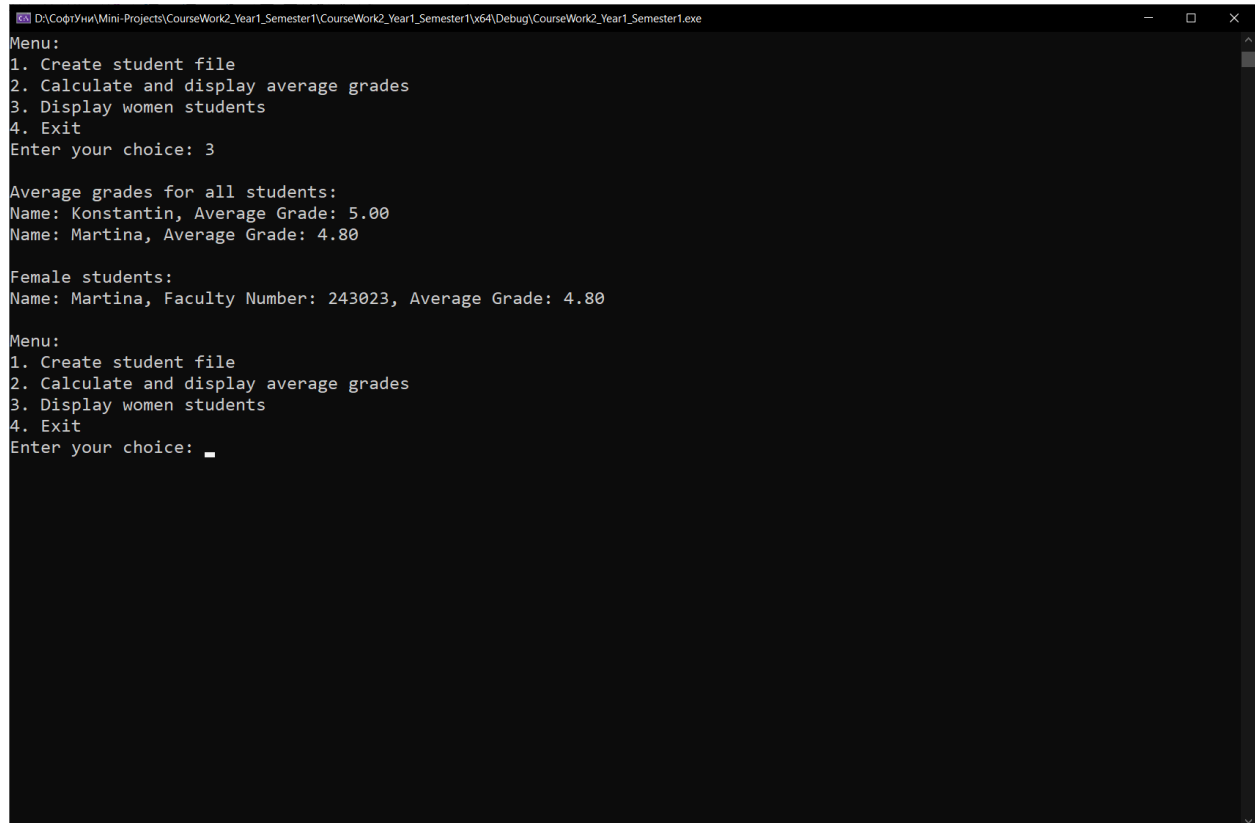
```
Select D:\Софтуър\Mini-Projects\CourseWork2_Year1_Semester1\CourseWork2_Year1_Semester1\Debug\CourseWork2_Year1_Semester1.exe
Menu:
1. Create student file
2. Calculate and display average grades
3. Display women students
4. Exit
Enter your choice: 2

Average grades for all students:
Name: Konstantin, Average Grade: 5.00
Name: Martina, Average Grade: 4.80

Menu:
1. Create student file
2. Calculate and display average grades
3. Display women students
4. Exit
Enter your choice:
```



### 3. *Display women students*



```
D:\Софт\Учи\Mini-Projects\CourseWork2_Year1_Semester1\CourseWork2_Year1_Semester1\Debug\CourseWork2_Year1_Semester1.exe
Menu:
1. Create student file
2. Calculate and display average grades
3. Display women students
4. Exit
Enter your choice: 3

Average grades for all students:
Name: Konstantin, Average Grade: 5.00
Name: Martina, Average Grade: 4.80

Female students:
Name: Martina, Faculty Number: 243023, Average Grade: 4.80

Menu:
1. Create student file
2. Calculate and display average grades
3. Display women students
4. Exit
Enter your choice:  
```