

---

# STUDNET MANAGEMANT SYSTEM

---

FINALE AUFGABE

GRUPPENMITGLIEDER

MARCEL ANDERS  
SIMON BAUR  
MARIAN MUTSCHLER

*Programmieren in C*  
*Frau Goertz*



2021

# Inhaltsverzeichnis

<b>1</b>	<b>Student Management System</b>	<b>2</b>
1.1	Allgemeines . . . . .	2
1.1.1	Intension / Aufgaenstellung . . . . .	2
<b>2</b>	<b>Functions</b>	<b>4</b>
2.1	Struct Student . . . . .	4
2.2	Struct Date . . . . .	4
2.3	void wait . . . . .	5
2.4	int getLength . . . . .	5
2.5	checkDate . . . . .	5
2.6	Function addStudent() . . . . .	5

# Kapitel 1

## Student Management System

### 1.1 Allgemeines

#### 1.1.1 Intension / Aufgaenstellung

- Erstelle eine Studenten Struktur mit folgenden Inhalten: Nachname, Matrikelnummer, Start Datum, End Datum, geburstdatum
- Erstelle eine Datums Struktur (day month year) fuer das gebursdatum und einschreibungsdatum etc.
- Schreibe eine Funktion inputStudent: Bei dem der Benutzer alle relevanten Daten zu einem Studenten eingeben kann.
- Schreibe eine Funktion addStudent: bei der ein Student hinzugefügt wird.
- Schreibe eine Funktion in der die Anzahl der gespeicherten Studenten zurueck gegeben werden soll.
- Schreibe eine Funktion printStudent(Matrikelnummer), mit der ein Student auf dem Bildschirm ausgegeben werden soll
- Schreibe eine Funktion printAllStudents, mit der alle Studenten Alphabetisch auf dem Bildschirm ausgegeben werden sollen.
- Schreibe eine Funktion menue, ueber die die Verschiedenen funktionen aufgerufen werden sollen.
- Schreibe eine funktion deleteStudent(matrikelnummer), welche einen Studenten loescht.
- Schreibe eine Funktion save die Alle gespeicherten Studenten in eine Datei speichert. Diese Funktion soll beim beenden des Programms automatisch aufgerufen werden und soll somit nicht im menue auftauchen

- Schreibe eine Funktion `read` die eine Datei ausliest und alle Studenten daraus ins Programm laed. Diese Funktion soll beim start des Programms automatisch aufgerufen werden und soll somit nicht im menue auftauchen
- Die Aufgabe soll mithilfe von Verketteten Listen gelöst werden.

## Kapitel 2

# Functions

This chapter contains each step of our project. Every functions, structs or somethings else can be found in the main.c. This is only a discription of the function.

### 2.1 Struct Student

First we create the structure for our students

```
struct student{
    char *surname;
    int matrikelnummer; //Muss 7 Zahlen lang sein
    struct date startdate;
    struct date exitdate;
    struct date birthdate;
    struct student *previous;
    struct student *next;
}*start=NULL, *end=NULL;
```

### 2.2 Struct Date

In our next step we create ihe struct date. This struct contains unsigned variables.

```
struct date{
    unsigned int day;
    unsigned int month;
    unsigned int year;
};
```

### **2.3 void wait**

Waiting for some user interaction.

### **2.4 int getLength**

Checking the length of a integer. A simple example: 10 = 2, 100=3. Also it checks the integer isn't equal 0 because in this case we get some errors.

### **2.5 checkDate**

Here we check the Date input. In case the input isn't a real date we return True.

### **2.6 Function addStudent()**

Here we have the function "addStudent". This function add the Student to the list. In our case this function only get pointers.