

# Imperative Programmierung

## Aufgabenblatt 10

---

**Hinweise:**

Bearbeitungszeit: 2 Wochen, Abgabefrist: Siehe Juniorstud.IP

Ladet eure Lösungen der Aufgaben unter "Aufgaben" im Juniorstud.IP als PDF-Dokument bzw.

C-Quelltext oder als txt-Datei hoch. Aufkommende Fragen bitte im Forum stellen, damit diese dort für alle geklärt werden können. Bei verspäteter Abgabe wird je Tag 10% der zu erreichenden Gesamtpunktzahl abgezogen.

---

1. Schreibt ein Programm, das eine Datei einliest, die zeilenweise Einträge der Form:

*Vorname Nachname Matrikelnummer Geburtsjahr*

enthält, und diese in umgekehrter Reihenfolge im Format:

*Matrikelnummer Nachname Vorname Geburtsjahr*

wieder ausgibt. So soll für eine Datei mit dem Inhalt:

```
John Doe 210123456 1991
Fred Feuerstein 211987501 1992
Erika Mustermann 213092495 1996
```

die Ausgabe:

```
213092495 Mustermann Erika 1996
211987501 Feuerstein Fred 1992
210123456 Doe John 1991
```

erfolgen.

Ihr benötigt für die Lösung dieser Aufgabe keine Arrays und keinen dynamischen Speicher. Für die Speicherung eines Eintrags ist die Nutzung einer `struct` sinnvoll. Ihr könnt davon ausgehen, dass Name und Vorname jeweils nicht länger als 20 Zeichen sind.

15 Punkte

2. Wie oft muss man im Durchschnitt ein Kartenspiel in die Luft werfen, bis es sortiert zu Boden fällt? Schreibt ein Programm, um das auszuprobieren. Ein Vektor der Länge  $N$ , der die Zahlen 1 bis  $N$  enthält, soll so lange zufällig durchmischt werden, bis die Zahlen sortiert im Vektor stehen. (Eine Zufallsdurchmischung eines Vektors könnt Ihr beispielsweise mit Hilfe des Fisher-Yates-Shuffle erzeugen.) Schätzt die mittlere Anzahl von Zufallsdurchmischungen, die benötigt werden, indem Ihr den Mittelwert von 1000 solcher Sortiervorgänge bestimmt. Gebt eine Tabelle der im Mittel benötigten Zufallsdurchmischungen für  $N = 2$  bis  $N = 10$  an. Welchen Zusammenhang vermutet Ihr?
- 25 Punkte