

# Übungsblatt Woche 13

## Präsenzaufgaben (25. - 27. Januar)

### Aufgabe 01 (\*\*\*)

- a) Erstellen Sie eine Struktur für Mitarbeiter. Die Struktur soll den Vor- und Nachnamen (je max. 20 Zeichen), das monatliche Gehalt (float) und die Abteilung (enum mit 3 Werten: Personal, Marketing, Leitung) des Mitarbeiters enthalten.
- b) Erstellen Sie eine Variable der Mitarbeiterstruktur mit plausiblen Werten („Max“, „Mustermann“, 4500.59, Personal).
- c) Implementieren Sie eine Funktion, die alle Eigenschaften eines Mitarbeiters auf der Kommandozeile ausgibt.
- d) Implementieren Sie eine Gehaltsanpassungsfunktion, welche einen Mitarbeiter und sein neues Gehalt als Parameter übergeben bekommt. Lassen Sie sich alle Informationen des Mitarbeiters vor und nach dem Aufruf der Funktion ausgeben.
- e) Implementieren Sie eine Versetzungsfunktion, mit der ein Mitarbeiter in eine andere Abteilung versetzt werden kann. Der Mitarbeiter und seine neue Abteilung sollen als Parameter an die Funktion übergeben werden. Sollte die alte und die neue Abteilung übereinstimmen, soll eine Fehlermeldung ausgegeben werden. Lassen Sie sich alle Informationen des Mitarbeiters vor und nach dem Aufruf der Funktion ausgeben.
- f) Implementieren Sie eine neue Funktion, welche die Änderung des Nachnamens eines Mitarbeiters erlaubt. Lassen Sie sich alle Informationen des Mitarbeiters vor und nach dem Aufruf der Funktion ausgeben. (Tipp: strcpy aus string.h).
- g) Erstellen Sie ein Array mit 4 Elementen vom Typ Mitarbeiter (mit sinnvollen Werten).
- h) Implementieren Sie eine Funktion, die ein Mitarbeiter-Array als Parameter übergeben bekommt und alle Informationen über Mitarbeiter ausgibt. Verwenden Sie hierzu die Funktion aus der Mitarbeiter-Aufgabe.
- i) Entfernen Sie den 2. Mitarbeiter aus dem Array. Wie könnte man das bewerkstelligen? Welche Probleme ergeben sich? Führen Sie b) nach dem Entfernen erneut aus.

- j) Entwerfen Sie eine Struktur, die einen Knoten in einem Binärbaum repräsentiert und einen Mitarbeiter beinhaltet. Legen Sie eine Funktion an, um einen Mitarbeiter in den Binärbaum einzufügen. Die Knoten sollen auf dem Heap liegen. Der Baum soll nach Gehalt sortiert sein (d.h., im linken Teilbaum eines Knotens sind alle Gehälter kleiner oder gleich, im rechten Teilbaum sind sie größer). Erstellen Sie in der main()-Funktion einen neuen Baum und fügen Sie Mitarbeiter mit zufälligen Gehälter hinzu.
- k) Schreiben Sie rekursive Funktionen, um den Baum sortiert auszugeben und um seinen Speicher freizugeben.

## Hausaufgaben (01.-03. Februar)

### Aufgabe 01 (\*\*\*)

- a) Erstellen Sie eine Struktur für 2D Integer Koordinaten (x und y). Schreiben Sie eine Funktion, die zwei Punktstrukturen als Parameter übergeben bekommt und den euklidischen Abstand zwischen den beiden Punkten berechnet und zurückgibt (Tipp: sqrt aus der math.h). Erstellen Sie zwei Variablen der Punktstruktur und rufen Sie die Abstandsfunktion damit auf. Lassen Sie sich das Ergebnis ausgeben.
- b) Erstellen Sie eine Struktur für Kreise. Die Struktur soll einen Mittelpunkt vom Typ der obigen Punktstruktur und einen Radius (int) enthalten. Implementieren Sie eine Funktion, die zwei Kreise als Parameter übergeben bekommt und diese auf Kollision überprüft. Verwenden Sie in dieser Funktion die Abstandsfunktion aus Aufgabe a). Die Funktion soll ein C-Äquivalent zu „wahr“ bzw. „falsch“ zurückgeben.
- c) Der Nutzer soll eine Anzahl n eingeben. Erstellen Sie ein dynamisches Array, das n Elemente vom Typ kreis enthalten kann. Füllen Sie dieses Array mit Kreisen. Die Daten dazu sollen zufällig erzeugt werden. Prüfen Sie anschließend alle Kreise paarweise auf Kollision und geben Sie entsprechende Meldung aus.