



Probeklausur zur Vorlesung  
**Programmierung**  
im Sommersemester 2019

Montag, den 15.07.2019

**Aufgabe 1. (10 Punkte)**



- (a) Allozieren Sie dynamisch Speicherplatz für ein Array von 20 Variablen des Typs `double`.
  
- (b) Erklären Sie Lebensdauer und Sichtbarkeit von Variablen. Gehen Sie hierbei kurz auf globale Variablen und `static` Variablen ein.
  
- (c) Definieren Sie eine Struktur (inklusive typedef) `student`, die ein Feld `name` des Typs `char[20]` und ein Feld `matrikelnummer` des Typs `int` besitzt. Deklarieren Sie eine Variable des Typs `student` und weisen Sie ihr Ihren Namen und Ihre Matrikelnummer zu.
  
- (d) Erklären Sie was man unter einem Buffer Overflow versteht.
  
- (e) Welche Vorteile ergeben sich bei der Speicherung von Daten in einer verketteten Liste im Vergleich zur Speicherung in einem Array?

**Aufgabe 2. (5 Punkte)**

Erklären Sie, wie mittels `getopt` Kommandozeilenargumente in C interpretiert werden können.

**Aufgabe 3. (6 Punkte)**

- (a) Schreiben Sie eine Funktion `random_double`, welche einen Parameter `double dmax` erhält und eine zufällige Zahl des Typs `double` im Bereich von 0 bis `dmax` zurückgibt.
  
- (b) Schreiben Sie eine Prozedur die eine `int n` Variable erhält und daraufhin rekursiv die Fakultät `n!` bestimmt.

**Aufgabe 4. (6 Punkte)**

- (a) Schreiben Sie eine Prozedur, die als Parameter einen String erhält und Großbuchstaben durch Kleinbuchstaben und umgekehrt ersetzt.
  
- (b) Schreiben Sie eine Prozedur, die als Parameter zwei Strings sowie ein `char` array und dessen Länge erhält. Hängen Sie den Inhalt des zweiten Strings an den ersten an und schreiben Sie das Ergebnis in das `char` array. Geben Sie eine Fehlermeldung aus, falls das Array zu klein ist.
  
- (c) Schreiben Sie eine Prozedur, die als Parameter ein `char` array, sowie zwei Strings erhält, die jeweils zwei Worte enthalten. Suchen Sie in dem array nach dem ersten Erscheinen des Inhalts des ersten Strings und ersetzen Sie ihn durch den Inhalt des zweiten. Sie dürfen annehmen, dass das Array groß genug ist.

**Aufgabe 5. (6 Punkte)**

- (a) Definieren Sie einen Datentypen `node`, welche die Node für eine einfach verkettete Liste repräsentiert und eine Struktur `tree_node`, welche die Node eines binären Baumes darstellt. Worin besteht der grundlegende Unterschied?

- (b) Eine Prozedur der Form

```
void print_tree_node(tree_node*);
```

sei bereits implementiert, welche den Inhalt einer einzelnen Node eines binären Baumes ausgibt. Schreiben Sie eine Prozedur, welche einen Pointer auf die Wurzelnode erhält und den Inhalt des binären Baumes rekursiv ausgibt. Die Reihenfolge ist hierbei egal, es sollen aber alle Noden ausgegeben werden.

**Aufgabe 6. (6 Punkte)**

- (a) Definieren Sie einen neuen Typen `employee`. Dieser soll durch eine Struktur gegeben sein, welche Felder für Namen, Vornamen und Gehalt enthält. Wählen Sie für die ersten zwei Felder jeweils ein `char` Array und für das Gehalt den Datentyp `int`.

- (b) Schreiben Sie eine Funktion

```
employee* new_employee(char* name, char* prename, int salary);
```

Diese soll Speicher für einen neuen Angestellten dynamisch allozieren und Namen, Vornamen und Gehalt gemäß der Parameter setzen.

- (c) Schreiben Sie eine Prozedur

```
void print_employee(employee*, char* filename);
```

welche einen Strom zu einer Datei mit dem Namen `filename` öffnet. Die Prozedur soll dann Namen, Vornamen und Gehalt formatiert an die Datei anhängen. Schließen Sie nach der Operation den Strom.