Systemnahe Software I
im Wintersemester 2009/2010

Abgabetermin: 09. Februar 2009

Dieses Übungsblatt ist freiwillig - jedoch eine gute Übung für die Klausur!
Versuchen Sie die ersten Aufgaben ohne Hilfe des Rechners zu lögsen!
Das Blatt zählt nicht zu den Übungspunkten hinzu, wird jedoch in der nächste Übung besprochen.

32 Preprocessor

Der C-Preprocesser wertet verschiedene Direktiven (= Anweisungen, Befehle) aus.
Nennen Sie 4!

32.1 Pointer

a) `int *(*x)();`
b) `int (*x[])();`
c) `int (*x())[];`

32.2 Deklarationen

Beschreiben Sie verbal, was hier genau deklariert wird:

```
int (*X) (double);
```

32.3 Obscure

```
int i1, i2, i3;
int *p1, *p2, *p3;

i2 = 12;
p2 = p3;
p1 = &i1;
*p2 = ((*p1)++)-1;
p3 = &i3;
--*p3;
*p3 -= 2 * (*p1 -*p2);
printf("%d %d %d\n", i1, i2, i3);
```

32.4 Code

Was geht bei diesem Programm schief?
Wie kann es verbessert werden?

```
char * test(char * txt) {
  char neu[100];
  char *x;

  x = neu;
  while ((*x++ = *txt++) != '\n'); /* kopieren */
  return(x);
}

int main () {
  char * strcp;

  strcp = test("Oh jeh!");
  printf("%s\n", strcp);
  return 0;
}
```

33 Bäume (7 Punkte)

Eine Möglichkeit zum Speichern abstrakter Daten sind Listen. Leider ist die Suche nach Datensätzen in grossen Listen sehr ineffizient. In den Vorlesungen AI1, AI2 haben Sie schon die Verwendung von Baumstrukturen kennengelernt.
Ein binärer Baum ist durch miteinander verknüpften Datensätzen (oft „Knoten“ genannt) gekennzeichnet. Jeder dieser Knoten enthält einen Namen sowie Referenzen auf bis zu zwei „Nachfolger“.

```
            Maria
            /   \
        Julia   Matthias
        /   \
   Gustav  Konrad
   /
 Adam
    \
     Berta
```

Ein solcher Baum lässt sich schrittweise von der „Wurzel“ aus („Maria“) aufbauen. Wie man sieht, befinden sich in den Teilbäumen, die an den Knoten jeweils links hängen, alphabetisch absteigende Namen. In den Knoten des rechten Teilbaumes nur alphabetisch aufsteigende Namen. Diejenigen Knoten, die keine Nachfolger mehr besitzen werden „Blätter“ genannt.

- Erstellen Sie ein Programm, welches einen solchen Binärbaum erstellt.
  Als erstes definieren Sie eine geeignete Datenstruktur für die einzelnen Knoten des Baumes.
  Den Namen in dieser Struktur können Sie dabei als Array of Char konstanter Länge festlegen.

- Fügen Sie eine Möglichkeit hinzu, den Baum nach In- Pre- und Postorder zu traversieren.
  Beispielsweise: (Eingefügte Namen: Maria - Matthias - Julia - Gustav - Konrad - Adam - Berta)
  Inorder (L-W-R): Adam - Berta - Gustav - Julia - Konrad - Maria - Matthias
  Preorder (W-L-R): Maria - Julia - Gustav - Adam - Berta - Konrad - Matthias
  Postorder (L-R-W): Berta - Adam - Gustav - Konrad - Julia - Matthias - Maria

- Erstellen Sie eine Funktion welche den Baum auch als solchen ausgibt:
  Beispielsweise: (Eingefügte Namen: Maria - Matthias - Julia - Gustav - Konrad - Adam - Berta)

```
  Matthias
Maria
    Konrad
 Julia
    Gustav
        Berta
      Adam
```

- Schreiben Sie zwei weitere Funktionen, um den Baum Laden bzw. Speichern zu können. Zum Speichern ist es ratsam, den Baum zu traversieren, und die Namen der Reihe nach in eine Datei zu schreiben.

Viel Erfolg!