

Klausur Einführung in die Programmierung II	Seite 1/7
Name, Vorname, Unterschrift:	Matrikelnummer:

Semestralklausur
Einführung in die Programmierung II
(Prof. Huckle)
Gruppe A
11. Juli 2007

Wichtig:

- Die Angabe besteht aus 7 Seiten — prüfen Sie bitte Ihr Exemplar auf Vollständigkeit.
- Kennzeichnen Sie jedes Blatt lesbar mit Ihrem Namen und dieses Deckblatt zusätzlich mit Ihrer **Matrikelnummer und Unterschrift**; trennen Sie die Heftung nicht auf.
- Bearbeiten Sie die Aufgaben auf den Angabenblättern. Sollte der Platz für einen Aufgabenteil nicht reichen, benutzen Sie bitte die Rückseite der Angabenblätter und machen Sie einen entsprechenden Vermerk bei der Aufgabe.
- Insgesamt können Sie 40 Punkte erreichen, bei jeder Aufgabe sind die für die jeweiligen Teilaufgaben vorgesehenen Punkte angegeben. Zum Bestehen sind 17 Punkte erforderlich.
- Als Hilfsmittel ist nur ein Ausdruck der Vorlesungsfolien erlaubt (handschriftliche Anmerkungen sind OK).
- Die Bearbeitungszeit beträgt 75 Minuten.

Name, Vorname:

1 Zeichenketten (4+4+7=15 Punkte)

a) Schreiben Sie eine Funktion mit einer Zeichenkette `s` als Parameter und Ergebnistyp `int`. Das Ergebnis soll 1 sein, wenn die Zeichenkette wenigstens ein `X` enthält (z.B. `"abcXdeX"`) und 0 sonst.

b) Schreiben Sie eine Funktion `ohneX` mit einer Zeichenkette `s` als Parameter. Die Funktion soll (mittels `printf` oder `std::cout`) die Zeichen von `s` der Reihe nach ausgeben — außer, das Zeichen ist ein `X` — die sollen ausgelassen werden.

Beispiel: `ohneX("abcXdefXghX")` druckt `abcdefgh`

Name, Vorname:

- c) Schreiben Sie nun eine Funktion `ersetze` mit zwei Zeichenketten `s` und `t` als Parameter. Die Funktion soll auch (mittels `printf` oder `std::cout`) die Zeichen von `s` der Reihe nach ausgeben — außer, das Zeichen ist ein `X`: in diesem Fall soll jetzt stattdessen das nächste Zeichen aus `t` ausgegeben werden, sofern es noch eins gibt (gibt es keins mehr, dann das `X`).

Zwei Beispiele:

- `ersetze("abcXefXhX", "DGI")` druckt `abcDefGhI`
- `ersetze("abcXefXhX", "DG")` druckt `abcDefGhX`

Name, Vorname:

2 Felder (6+9=15 Punkte)

In dieser Aufgabe betrachten wir $n \times n$ -Matrizen der Form

$$A = \begin{pmatrix} b & c & & & \\ a & b & c & & \\ & a & b & \ddots & \\ & & \ddots & \ddots & c \\ & & & a & b \end{pmatrix}$$

(d.h., alle Diagonalelemente sind b , alle Komponenten der unteren Nebendiagonale sind a , alle der oberen Nebendiagonale sind c , alle übrigen Komponenten sind Null), wobei a , b und c ganze Zahlen sind.

- a) Schreiben Sie eine Funktion mit der Matrixgröße n und den Koeffizienten a , b und c als Parameter, die die Komponenten der Matrix ausdrückt (mittels `printf` oder `std::cout`). Es sollen alle n^2 Werte gedruckt werden (also auch die Nullen), eine Matrixzeile soll jeweils auf einer Druckzeile stehen und die Komponenten sollen in der natürlichen Anordnung (von oben nach unten und von links nach rechts) gedruckt werden; weitere Bedingungen an die Formatierung gibt es nicht (die Komponenten einer Spalte brauchen also nicht exakt untereinander zu stehen).

Name, Vorname:

- b) Schreiben Sie eine Funktion, die die Matrix A mit einem Vektor x mit ganzzahligen Komponenten multipliziert.

Der Komponenten von x seien in der natürlichen Reihenfolge in einem Feld von **int** gespeichert, die Funktion bekommt dieses Feld als Parameter sowie dessen Länge n und die drei Werte a , b und c .

Für den Ergebnisvektor $y = A \cdot x$ soll mittels `malloc` oder `new` ein Feld bereitgestellt werden; das Funktionsergebnis soll ein Zeiger auf das erste Element dieses Feldes sein.

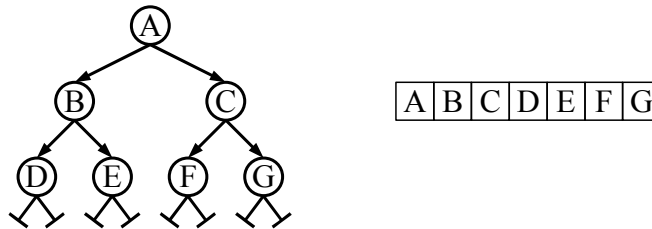
Name, Vorname:

3 Binärbäume (10 Punkte)

Ein Binärbaum heißt *vollständig*, wenn alle Blätter die gleiche Tiefe t haben und alle Knoten, die kein Blatt sind, zwei Söhne haben.

Im Bild ist das leichter zu verstehen: ein vollständiger Binärbaum sieht aus wie im linken Teil der Skizze gezeichnet (da ist die Tiefe $t = 3$).

Für Tiefe t hat ein vollständiger Binärbaum $2^t - 1$ Knoten, die Beschriftungen der Knoten (die hier vom Typ `char` seien) kann man daher bequem in einem Feld mit $2^t - 1$ Komponenten verpacken. Dies soll hier „schichtweise“ passieren, bei der Wurzel beginnend und bei den Blättern endend. Der Baum in der Skizze links wäre dann wie in der Skizze rechts gespeichert.



Schreiben Sie eine Funktion `rechts` mit Parameter k und Ergebnis vom Typ `int`, die zu dem Knoten mit Feldindex k den Feldindex des rechten Sohnes berechnet (in der Skizze hat z.B. der Knoten C den Index 2 und sein rechter Sohn G den Index 6, also soll das Ergebnis von `rechts(2)` der Wert 6 sein):

Im Folgenden sei angenommen, dass es analog eine Funktion `links` für den Index des linken Sohns gibt (`links(2)` wäre z.B. 5) und eine Funktion `int nknoten(int t)`, die $2^t - 1$ berechnet.

Für die, die es ganz genau nehmen: über den Zeichensatz unseres Systems sei angenommen, dass die Buchstaben in der normalen Reihenfolge codiert sind: A, dann B, dann C,...

Name, Vorname:

Nun werden folgende drei Funktionen betrachtet:

```
void fuellen_a(char *b, int t, int k, char x) {
    if (t>0) {
        fuellen_a(b, t-1, links(k), x+1);
        b[k] = x;
        fuellen_a(b, t-1, rechts(k), x+1);
    }
}

void drucken_a(char *b, int t) {
    int i;
    int n = nknoten(t);
    for (i=0; i<n; i++)
        printf("Knoten %d: %c\n", i, b[i]);
    // C++: std::cout << "Knoten " << i << ": " << b[i] << '\n';
}

int main() {
    int t=3;
    char *b = (char *)malloc(nknoten(t)*sizeof(char));
    // C++: char *b = new char[nknoten(t)];
    fuellen_a(b, t, 0, 'A');
    drucken_a(b, t);
}
```

Was wird ausgedruckt, wenn das Programm (also die Funktion `main`) ausgeführt wird?