

7. Nullstellen [8 Punkte]

Implementieren Sie eine Funktion

```
double zero(double(*f)(double), double a, double b);
```

die unter den Voraussetzungen:

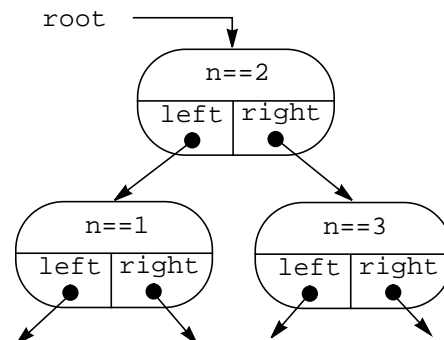
1. f ist stetig im Intervall $[a, b]$ und
2. $f(a) * f(b) < 0$

eine Nullstelle der Funktion f näherungsweise bestimmt.

8. Sortieren mit Binären Bäumen [10 Punkte]

Schreiben Sie eine C++-Programm, welches eine Folge von eingelesenen ganzen Zahlen (die mit 0 beendet sein soll) sortiert ausgibt. Verwenden Sie dazu einen binären Baum, dessen Knoten Objekte der folgenden Klasse sind:

```
class BTreeNode {  
    int n;  
    BTreeNode * left, *right;  
    static BTreeNode* root;  
    void insert (BTreeNode* & = root);  
public:  
    BTreeNode(int);  
    static void out(BTreeNode* = root);  
};
```



Nach dem Einlesen einer Zahl ist jeweils ein neuer Baumknoten zu erzeugen und in den bisherigen Baum einzufügen, wobei in jedem Teilbaum links nur Elemente kleiner/gleich dem Wurzelement und rechts Elemente größer dem Wurzelemente enthalten sind. Die Operation `BTreeNode::out()`; soll abschließend den Inhalt des gesamten Baumes geordnet ausgeben.

(*) 9. Zugabe [4 Punkte]

Das folgende Programm läßt sich (überraschenderweise) fehlerfrei übersetzen. Was gibt es aus ?

```
#include <iostream.h>  
  
main()  
{  
    cout<<1["]<<2["]<<endl;  
}
```

```
typedef time_t long;
```

benutzt, die die Anzahl der seit dem 1. Januar 1970, 0:00 Uhr vergangenen Sekunden als Wert enthält. Schreiben Sie ein Programm, welches den exakten Zeitpunkt bestimmt, zu dem diese „Uhr überläuft“, wenn (wie meist üblich) `sizeof(long)==4` gilt.

Hinweis: In der C-Bibliothek gibt es eine Standardfunktion

```
char* ctime(const time_t*);
```

die einen Wert des Typs `time_t` in eine lesbare Zeichenkette der Form

```
Fri Feb 7 13:17:44 1997
```

umwandelt.

6. Fehlertext [10 Punkte]

Finden Sie möglichst viele Fehler in dem folgenden vermeintlichen C++-Programm, welches nach Eingabe einer (mit 0 abgeschlossenen) Folge von Noten (1..6) einen „Zensurenspiegel“ der Form

```
1 | 2 | 3 | 4 | 5 | 6 | (/  
4 | 5 | 3 | 3 | 2 | 1 | 2.83333
```

ausgeben soll. Jede Note komme dabei höchstens neunmal vor. Falsche Eingaben sollen ignoriert werden.

```
#include "iostraem.h"
```

```
int main()  
{  
    cout<<" Bitte Noten eingeben: \n(Ende: Eingabe von 0)\n\n";  
  
    int n=0; sum; count;;  
    do {  
        cin >> n;  
        if (n<1 && n>6) continue;  
        sum+=n;  
        noten[n]++;  
        cout++;  
  
        while (n);  
  
        double mean = sum / count;  
  
        for (int i=1; i<=6; i++);  
            cout << i << " | ";  
        cout << " (/\n"  
        for (int j=0, j<6, j++)  
            cout << noten[j] < " | ";  
        cout << mean << '\n'; /* und fertig !  
    }  
  
    int *noten = new int [6];
```

3. Konstruktoren / Destruktoren [8 Punkte]

Welche Ausgaben erzeugt das folgende C++-Programm ?

```
#include <iostream.h>

class X {
    static void p(const char* s) { cout<<"X::"<<s<<endl; }
public:
    X()          {p("X()");}
    X(int)       {p("X(int)");}
    X(const X&)  {p("X(constX&)");}
    ~X()        {p("~X()");}
};

X x1, x2=2, x3=x2;

X x4();

X f (X x) {return x;}
X& g (X& x){return x;}

int main()
{
    cout<<"start\n";
    X x5 (2);
    X* px=new X;
    {
        x1=f(x5);
    }
    x2=g(*px);
    cout<<"end\n";
}
```

4. Römische Zahlen [10 Punkte]

Schreiben Sie eine Funktion `int r2i (const char*)`, die eine (korrekte) Römische Zahl (welche als Zeichenkette repräsentiert ist) in die entsprechende Dezimalzahl umwandelt. Bei den Grundzeichen der Römischen Zahl soll dabei Groß- und Kleinschreibung zulässig sein (M = m, ..., I = i).

Hinweis: Römische Zahlen bestehen aus den Grundzeichen M (=1000), C (=100), X (=10) und I (=1) und den Hilfszeichen D (=500), L (=50) und V (=5). Zur Darstellung einer Zahl werden die Zeichen aus denen sich diese Zahl additiv zusammensetzt in numerisch absteigender Folge notiert. Grundzeichen können mehrfach auftreten, Hilfszeichen höchstens einfach. Zur Abkürzung der Notation wird höchstens ein vorgestelltes Grundzeichen benutzt, um „Vierer-“ bzw. „Neuner“-Werte darzustellen: XLIV = 44, IC = 99, XM = 990.

5. Time goes by [6 Punkte]

Derzeit wird in vielen UNIX-Betriebssystemen zur Darstellung der absoluten Zeit eine Variable des Typs `time_t`

1. Klausur

zur Vorlesung

„Objektorientierte Programmierung in C++“

(Abschluß 1. Semester [elementares C++]) 1996/97

Bitte notieren Sie auf allen Lösungsblättern Ihren Namen. Die Bekanntgabe der Ergebnisse und die Diskussion von Beispiellösungen erfolgt am 14.2.97.

Mit der vollständigen Lösung der folgenden 8 Aufgaben können insgesamt 68(+4) Punkte erreicht werden. Streben Sie bei der Auswahl der von Ihnen gelösten Aufgaben 34 Punkte an.

1. Primzahlen [6 Punkte]

Schreiben Sie ein C++- Programm, welches beim Aufruf mittels

```
% prim n <enter>          n sei dabei eine natürliche Zahl > 0
```

die ersten n Primzahlen bestimmt und (via Standardausgabe) ausgibt.

Hinweis: In der C-Standardbibliothek gibt es eine Funktion `int atoi (const char*)`; , die eine Zeichenkette in die repräsentierte Dezimalzahl umwandelt.

2. Dreiecke [8 Punkte]

Implementieren Sie eine Klasse `Dreieck`. Mit Objekten dieser Klasse sollen folgende Operationen möglich sein:

- Vereinbarung von Objekten durch Angabe von 3 Seitenlängen:

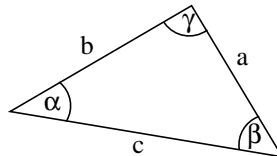
```
Dreieck d (3.0, 4.0, 5.0);
```

- Test, ob die angegebenen Seitenlängen tatsächlich ein Dreieck bilden:

```
if (d.check()) ...
```

- Berechnung von Umfang, Fläche und Innenwinkeln eines Dreiecks:

```
double u = d.umfang();  
double a = d.flaeche();  
double alpha = d.alpha();  
double beta  = d.beta();  
double gamma = d.gamma();
```



Hinweise: Man verwende die sog. *Heronische Dreiecksformel*

$$A = \sqrt{s \cdot (s - a) \cdot (s - b) \cdot (s - c)} \quad , \quad s = u / 2 \quad \text{den Kosinussatz}$$

$$c^2 = a^2 + b^2 - 2ab \cdot \cos \gamma \quad \text{und ggf.}$$

```
const double pi=3.14159265358979323846;
```

In der C-Standardbibliothek gibt es Funktionen

```
double sqrt(double); // square root und  
double acos(double); // arcus cosinus im Bogenmaß
```