

Angewandte Mathematik und Programmierung

Einführung in das Konzept der objektorientierten Anwendungen zu
mathematischen Rechnens

SS2013

Fomuso Ekellem



Inhalt

- **Projekt Vorlesung: praktische Implementierung üben**
 - Ein und Ausgabe usw.



Aufgabe 1

- Ein Zeichen soll eingelesen und dann wieder ausgegeben werden. Dabei soll die Ausgabe sowohl als Zeichen als auch als Dezimal- bzw. Hexadezimalwert erfolgen (z.B.: `z->z 122 0x7A`).



Lösung 1

```
# include <iostream>
using namespace std ;
int main () {
char z;
cout << "Ein Zeichen eingeben : ";
cin >> z;
cout <<"Das Zeichen ist : " <<z <<endl <<"Dezimal : "<< int(z)<<endl
<<"Hex : 0x"<<hex << int(z)<<endl;
system("pause");
return 0;

}
```



Aufgabe 2

- Erstellen Sie ein Programm, das mithilfe einer for-Schleife alle durch 7 teilbaren Zahlen zwischen zwei zuvor eingegebenen Grenzen ausgibt.



Lösung 2

```
# include < iostream >
using namespace std ;
int main ( void )
{
int a , b;
cout << " Untergrenze : "; cin >> a;
cout << " Obergrenze : "; cin >> b;

for ( int i=a; i <=b; i=i+1)
if (i %7==0) cout << i << " ";
system("pause");
return 0;
}
```



Aufgabe 3

- Schreiben Sie ein Programm, das zu einem gegebenen Anfangskapital bei einem vorgegebenem jährlichen Zinssatz berechnet, wie viele Jahre benötigt werden, damit das Kapital eine bestimmte Zielsumme überschreitet.



Lösung 3

```
# include < iostream >
using namespace std ;
int main ( )
{
    int jahre =0;
    double kapital , zins , endkapital ;
    cout << " Anfangskapital : ";
    cin >> kapital ;
    cout << " Zinssatz in %: ";
    cin >> zins ;
    cout << " Angestrebtes Endkapital : ";
    cin >> endkapital ;
```

```
while ( kapital < endkapital )
{
    jahre ++;
    kapital += kapital *( zins / 100);
}
cout << " Es dauert " << jahre << " Jahre
    !" << endl ;
system("pause");
return 0;
}
```




Aufgabe 4

- **Geben Sie alle Primzahlen zwischen 2 und einer vom Benutzer einzugebenden Obergrenze aus. Bestimmen Sie die Primzahlen mittels Ganzzahldivision.**



Lösung 4

```
# include < iostream >
using namespace std ;
int main ( void )
{
    int zahl , teiler , limit ;
    cout << " Obergrenze eingeben : ";
    cin >> limit ;
    cout << "\ nPrimzahlen bis " << limit << ":\n";
        for ( zahl =2; zahl < limit ; zahl ++ )
        {
            teiler = 2;
            while ( zahl % teiler != 0) teiler ++;
            if ( teiler == zahl ) cout << zahl << " ";
        }
    cout<<endl;
    system("pause");
    return 0;
}
```



Aufgabe 5

- **Christbaum‘ durch Ausdruck entsprechend vieler Leerzeichen und Sterne auf dem Bildschirm aus.**



Lösung 5

```
# include < iostream >
using namespace std ;
int main ( void )
{
    int blzahl =20 , sternzahl =1 , i , zeile ;
    for ( zeile =0; zeile <10; zeile ++ )
    {
        for (i =0; i< blzahl ; i ++ ) cout <<" ";
        for (i =0; i< sternzahl ; i ++ ) cout << "*";
        blzahl --;
        sternzahl +=2;
        cout << endl;
    }
    system("pause");
    return 0;
}
```



Aufgabe 6

In unserem Kalender sind zum Ausgleich der astronomischen und kalendarischen Jahreslänge in regelmäßigen Abständen Schaltjahre eingebaut. Zur exakten Festlegung der Schaltjahre dienen die folgenden Regeln:

- ☐ Ist die Jahreszahl durch 4 teilbar, so ist das Jahr ein Schaltjahr. Diese Regel hat allerdings eine Ausnahme:
- ☐ Ist die Jahreszahl durch 100 teilbar, so ist das Jahr kein Schaltjahr. Diese Ausnahme hat wiederum eine Ausnahme:
- ☐ Ist die Jahreszahl durch 400 teilbar, so ist das Jahr doch ein Schaltjahr.

Erstellen Sie ein Programm, das berechnet, ob eine vom Benutzer eingegebene Jahreszahl ein Schaltjahr bezeichnet oder nicht.



Lösung 6

```
# include <iostream>
using namespace std;
int main ()
{
    int jahr ;
    cout<<" Geben Sie ein Jahr ein : ";
    cin>>jahr ;
    if ( jahr %4==0)
    if( jahr %100==0)
    if( jahr %400==0)
    cout<<" Schaltjahr !"<<endl;
    else
    cout<<" Kein Schaltjahr!"<<endl;
    else
    cout<<" Schaltjahr!"<<endl;
```

```
else
    cout<<" Kein Schaltjahr!" <<endl;

    system("pause");
    return 0;
}
```



Aufgabe 7

Programmieren Sie das Spiel '

- **Zahlenraten': Der Computer generiert eine Zufallszahl zwischen 1 und 100, die der Spieler erraten muß. Es wird bei jedem Durchgang mitgeteilt, ob die eingegebene Zahl zu groß oder zu klein war.**
- **Tipp: Zufallszahlen werden wie folgt generiert: Ein einmaliger Aufruf `srand(time(NULL))` initialisiert den Zufallszahlengenerator. Anschließend liefert jeder Aufruf von `rand()%100` eine Zufallszahl zwischen 0 und 99.**
- **Die Funktionen sind in `stdlib.h` bzw. `time.h` deklariert.**



Lösung 7

```
/ # include < stdio.h>
# include < time.h>
# include < stdlib.h>
#include <iostream>
using namespace std;
int main ( void )
{
    int zahl , rat , versuche =0;

    srand ( time ( NULL ));
    zahl = rand ()%100 + 1;
```

```
do
{
    cout<<" Zahl eingeben : ";
    cin>> rat;
    if ( rat > zahl ) cout<<" Zu
        gross !\"<<endl;
    if ( rat < zahl ) cout<<" Zu
        klein !\"<<endl;
    versuche ++;
}
```

```
while ( rat != zahl &&
        versuche <6);

if ( rat == zahl ) cout<<"
    Erraten !\"<<endl;
else cout<<" Verloren ! Die
    gesuchte Zahl war"
    <<zahl<<endl;
    system("pause");
    return 0;
}
```




Aufgabe 8

- **Programmieren Sie mithilfe der switch-case Anweisung einen Mini-Taschenrechner, der zwei Zahlen und einen Operator (+, -, *, /) einliest und das Ergebnis ausgibt. Eine eventuelle Division durch 0 soll abgefangen werden.**



Lösung 8

```
using namespace std ;
int main ( void )
{
    float z1 ,z2;
    char op;
    cout << "2 Zahlen mit Operator
        dazwischen [+ -*/]: ";
    cin >> z1 >> op >> z2;
    cout << z1 << op << z2 << "=";
    switch ( op)
    {
        case '+':
            cout << z1+z2 << endl;
            break ;
        case '-':
```

```
        cout << z1-z2 << endl;
        break ;
        case '*':
            cout << z1*z2 << endl;
            break ;
        case '/':
            if ( z2 ==0)
                cout << " Div durch 0 geht nicht \n";
            else
                cout << z1/z2 << endl;
            break ;
        default :
            cout << " Den Operator kenne ich nicht \n";
    }

    system("pause");
    return 0;
```



Aufgabe 9

- Erstellen Sie ein Programm, das einen Text aus einer Datei einliest und auf dem Bildschirm ausgibt, wobei Groß- in Kleinbuchstaben verwandelt werden sollen und umgekehrt.



Lösung 9

```
# include < iostream >
# include < fstream >

using namespace std ;

int main (void)
{
    char c;
    ifstream datei ;

    datei.open ("demo.txt", ios::in );
    if (!datei )
    {
        cout << "Datei nicht gefunden !\n";
        return 1;
    }
}
```

```
while ( datei.get (c))
{
    if (c>='a' && c <='z') c=c+('A'-'a');
    else if (c >='A' && c <='Z') c=c -('A'-'a');
    cout << c;
}

cout<<endl;
system("pause");
datei.close ();
return 0;
}
```



Aufgabe 10

- Schreiben Sie eine Funktion `stringlength`, an die ein `char`-Array übergeben wird, und die die Länge dieser Zeichenkette zurückliefert.
- Tipp: Das Ende einer Zeichenkette ist durch `'\0'` (ASCII: 0) gekennzeichnet.



Lösung 10

```
# include < iostream >
using namespace std ;
```

```
int stringlength ( char *w);
int main ( void )
{
char wort [50];
```

```
cout << " Wort eingeben : ";
cin >> wort ;
cout << " Das Wort " << wort <<" hat "
<< stringlength ( wort ) << " Zeichen ." << endl ;
system("pause");
return 0;
}
```

```
int stringlength ( char *w)
{
int i=0;

while (w[i]!='\0 ') i++;
return i;
}
```