

Übungen zu C++ im Wintersemester 2014
Aufgabenblatt 7

Aufgabe 18:

Wandeln Sie das BCPL-Programm „Freitag, der 13-te“ in ein C++-Programm.

```
GET "libhdr"
```

```
MANIFEST { mon=0; sun=6; jan=0; feb=1; dec=11 }
```

```
LET start() = VALOF
  {LET count      = TABLE 0, 0, 0, 0, 0, 0, 0
   LET daysinmonth = TABLE 31,  ?, 31, 30, 31, 30,
                           31, 31, 30, 31, 30, 31

   LET days = 0
   FOR year = 1973 TO 1973+399 DO
     {daysinmonth!feb := febdays(year)
     FOR month = jan TO dec DO
       { LET day13 = (days+12) REM 7
        count!day13 := count!day13 + 1
        days := days + daysinmonth!month
       }
     }
   FOR day = mon TO sun DO
     writef("%i3 %sdays*n",
            count!day,
            select(day,
                  "Mon", "Tues", "Wednes", "Thurs",
                  "Fri", "Sat", "Sun"))
   RESULTIS 0
}
```

Aufgabe 19:

Schreiben Sie ein C++-Unterprogramm der Signatur

```
vector<char> wandlungb (int zahl, int ziffernzahl)
```

das die Darstellung von echten Dezimalbrüchen im Dualsystem als Folge von 0 und 1 bestimmt, hierbei sind zahl die Ganzzahl hinter dem Dezimalkomma und ziffernzahl die Anzahl der Ziffern von zahl.

Berechnen Sie die Bitketten zu den Zahlen 0,3125 und 0,7812510, die Periode für den Dualbruch der zweiten Zahl ist größer als 1000.

Beispiel: Wandlung von 0,3 in Dualdarstellung

$$0,3 * 2 = 0,6 \quad \Rightarrow 0$$

$$0,6 * 2 = 1,2 \quad \Rightarrow 1$$

$$0,2 * 2 = 0,4 \quad \Rightarrow 0$$

$$0,4 * 2 = 0,8 \quad \Rightarrow 0$$

$$0,8 * 2 = 1,6 \quad \Rightarrow 1$$

$$0,6 * 2 = \dots$$

Damit $0,3 = 0,01001\dots$

Vorperiode: Länge = 1, Ziffernfolge = 0

Periode: Länge = 4, Ziffernfolge = 1001

Aufgabe 20:

Nutzen Sie das Programm aus Aufgabe 19, um die Rundungsregel bei der Ausgabe für Dezimalbrüche und Sedezimalbrüche zu bestimmen. Für Sedezimalbrüche nutzen Sie den Manipulator hexfloat.