

Lösung der Aufgabenserie 5 Grundlagen der Informatik und Numerik

1. Speicher

1 Seite = 2 * 60 Anschläge pro Zeile * 40 Zeilen = 4800 B (Bytes)
 1.44 MB = 1024 KB + 450.56 KB = 1474.56 KB
 = 1 509 949,44 B
 = 314,5728 Druckseiten
 700 MB = 716800 KB = 734 000 000 B = 152 917.33 Druckseiten
 1 GB = 1024 MB = 1 048 576 KB = 1 073 700 000 = 223 696.21 Druckseiten

2. Elementardatentypen (Maschinenzahlen)

(a) Zweierkomplement

10011101 - 1 = 10011100
 $\frac{1}{0} \frac{0}{1} \Rightarrow -(1100011)_2 = -(2^6 + 2^5 + 2^1 + 2^0) = -(64 + 32 + 2 + 1) = -99$
 $32\ 368 = 2^{15} \Rightarrow \text{int-Zahl} \Rightarrow 0000\ 0000\ 0000\ 0000\ 1000\ 0000\ 0000\ 0000$

(b) IEEE-Code

$11.75 = (1011.11)_2 = (1.0111\ 1)_2 * 2^3 = (-1)^0 (1.0111\ 1)_2 * 2^{130-127}$
 Maschinenzahl: 0|100 0001 0|011 1100 0000 0000 0000 0000
 $-5.0 = -(101.0)_2 = -(1.01)_2 * 2^2 = (-1)^1 * (1.01)_2 * 2^{129-127}$
 Maschinenzahl: 1|100 0000 1|010 0000 0000 0000 0000 0000

(c) **float:** $x = 13.33308315277099609375$ $y = -0.375$
int: $x = 1096111183$ $y = -1094713344$
ASCII: $x = \text{„AUTO“}$

(d) $x = 2^{22} = 1 * 2^{149-127}$
 149 : 2 = 74 R 1
 74 : 2 = 37 R 0
 37 : 2 = 18 R 1
 18 : 2 = 9 R 0 $\Rightarrow x' = 0\ 10010101\ 0 \dots 0$ (23 Nullen)
 9 : 2 = 4 R 1
 4 : 2 = 2 R 0
 2 : 2 = 1 R 0
 1 : 2 = 0 R 1

$y = 2^{-3} = 1 * 2^{124-127}$
 124 : 2 = 62 R 0
 62 : 2 = 31 R 0
 31 : 2 = 15 R 1
 15 : 2 = 7 R 1 $\Rightarrow y' = 0\ 01111100\ 0 \dots 0$
 7 : 2 = 3 R 1
 3 : 2 = 1 R 1
 1 : 2 = 0 R 1

$z = 2^{22} + 2^{-3} = 100'0000'0000'0000'0000'0000.001 = 1.00'0000'0000'0000'0000'001 * 2^{22}$
 $\Rightarrow z' = x' = 0\ 10010101\ 0 \dots 0$

3. Primzahlbestimmung (Algorithmen und Programme)

(a) Gegenbeispiel: $n = 41$

(b) Programm

$$p = \text{prim}(n) \begin{cases} = 0, & \text{falls } n \text{ keine Primzahl} \\ \neq 0, & \text{sonst} \end{cases}$$

```

p = 0;
if( n > 1)
{
    if( n == 2)
    {
        p = 1;           // 2 ist Primzahl
    }
    else
    {
        t = 2; p = n % t;
        if( p != 0)     // n ist ungerade
        {
            a = n / t; t = 3;
            while( a > t) // n >= t * t
            {
                p = n % t;
                if( p != 0)
                {
                    p = 1; a = n / t; // Abbruchkriterium
                    t = t + 2;
                }
                else
                {
                    a = t; // erzwungener Abbruch
                }
            }
        }
    }
}

```

(c) **Protokoll**

<i>Durchlauf</i>	0	1	2	3	4	5
<i>n = 31</i>	<i>a =</i>	31	15	10	7	6
	<i>t =</i>	2	3	4	5	6
	<i>p =</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>

⇒ *prim(31) = true*, Abbruch, da Schleifenbedingung nicht mehr erfüllt.

<i>Durchlauf</i>	0	1	2	3	4	5	6
<i>n = 623</i>	<i>a =</i>	623	311	207	155	124	103
	<i>t =</i>	2	3	4	5	6	7
	<i>p =</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>false</i>

⇒ *prim(623) = false*, erzwungener Abbruch, 7 ist Teiler von 623.

Programm in C:

prim.c

```

/*
 * Primzahltest
 */
# include <stdio.h>

int main()
{
    int p;           /* 1 .. Primzahl, sonst 0 */
    int n;           /* Testzahl */
    int a;           /* n / ( t - 1) */
    int t;           /* potentieller Teiler */

    printf( "Primzahltest von n = ");      /* Eingabe */

```

```
scanf( "%d", &n);

/* ----- Beginn Programm Modellrechner ----- */
p = 0;
if( n > 1)
{
    if( n == 2)
    {
        p = 1;                /* 2 ist Primzahl */
    }
    else
    {
        t = 2; p = n % t;
        if( p != 0)          /* n ist ungerade */
        {
            a = n / t; t = 3;
            while( a > t)    /* n >= t * t */
            {
                p = n % t;
                if( p != 0)
                {
                    p = 1; a = n / t;    /* Abbruchkriterium */
                    t = t + 2;
                }
                else
                {
                    a = t;                /* erzwungener Abbruch */
                }
            }
        }
    }
}
/* ----- Ende Programm Modellrechner ----- */

if( p)                /* Ergebnis des Tests */
    printf( "%d ist Primzahl!", n);
else
    printf( "%d ist keine Primzahl, %d ist Teiler", n, t);

return 0;
}
```

4. Projekt Arbeitsblätter, Arbeitsmaterial zur Planimetrie

siehe Netz