

# Informatik 1 für Regenerative Energien

## Klausur vom 7. Juli 2014: Lösungen

Jörn Loviscach

Versionsstand: 7. Juli 2014, 16:37



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

1. 13 ist  $00001101_2$  und  $-6$  ist  $11111010_2$ .

$$\begin{array}{r} 00001101 \\ +11111010 \\ \hline 11111 \\ \hline 00000111 \end{array}$$

2.  $\sim a$  ist  $0x5E$ ,  $a \& b$  ist  $0x21$ ,  $a^b$  ist  $0x94$ .

3.  $((a \% 10) \neq 3) \parallel ((! c) \&\& ((b / 100) \leq 1))$
- |                 |                 |                |
|-----------------|-----------------|----------------|
| _____3_____     | false           | _____1_____    |
| _____false_____ |                 | _____true_____ |
|                 | _____false_____ |                |
| _____false_____ |                 |                |

4. `bool istSchnapszahl(unsigned int z)`

```
{
    if(z < 10) // < statt >
    {
        return false;
    }

    int letzteZiffer = z % 10; // % statt /
    while(z > 0)
    {
        if(z % 10 != letzteZiffer) // != statt ==
        {
            return false;
        }
        z /= 10;
    }
    return true;
}
```

## 5. Zum Beispiel so:

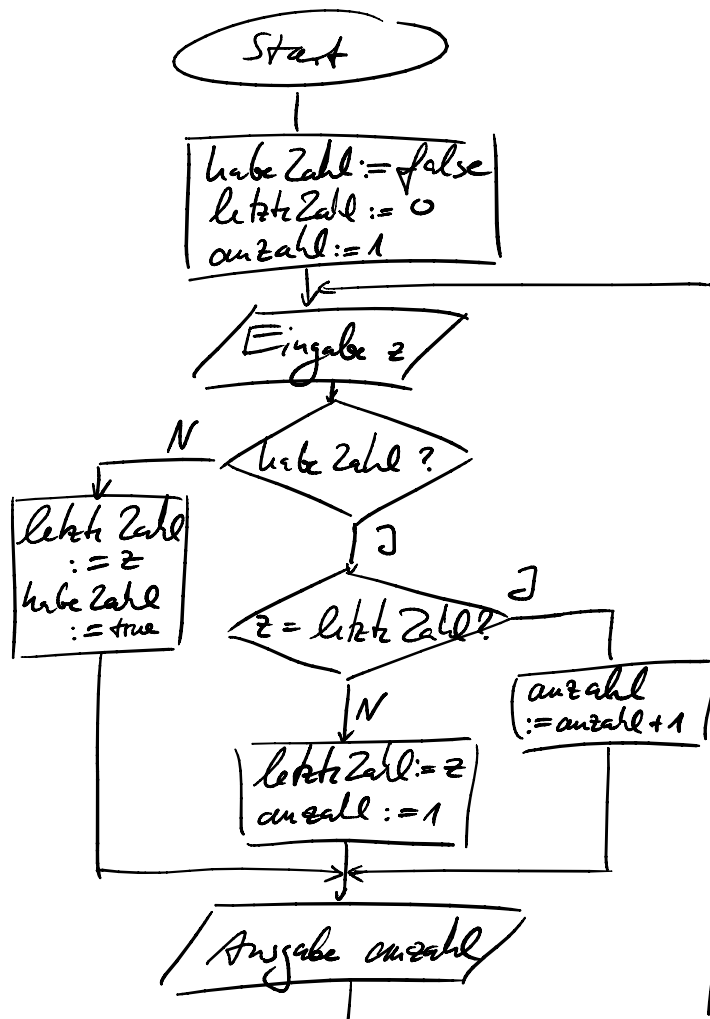
```

#include <math.h>

double berechneMaximalenAbstand(Punkt punkte[], int anzahl)
{
    double maxAbst = 0.0;
    for(int i = 0; i < anzahl; i++)
    {
        double x = punkte[i].x;
        double y = punkte[i].y;
        double abst = sqrt(x*x + y*y);
        if(abst > maxAbst)
        {
            maxAbst = abst;
        }
    }
    return maxAbst;
}

```

## 6.



## 7. Zum Beispiel so:

```

bool pruefeEMailAdresse(char s[])
{
    bool habeAt = false;
    bool habePunktNachAt = false;
    for(int i = 0; s[i] != 0; i++)
    {
        if(s[i] == '@')
        {
            if(habeAt) // ein zweites @ gefunden
            {
                return false;
            }
            habeAt = true;
        }
        else if(habeAt && s[i] == '.')
        {
            habePunktNachAt = true;
        }
    }
    return habePunktNachAt;
}

```

8. *Das erste if wird durch die beiden if darin überflüssig!*

```

switch(x)
{
    case 5:
        y = 7;
        break;
    case 9:
        y++;
        break;
    case 3:
        y = 2;
        break;
}

```

## 9. 9

*Es ist nicht das Ergebnis der Funktion gefragt, sondern der Wert von a!*

10. 7, 0, 6

11. 256, -1

```

12. int f_neu(int a)
{
    return 13;
}

```