

# Informatik 2 für Regenerative Energien

## Klausur vom 7. Juli 2011: Lösungen

Jörn Loviscach

Versionsstand: 13. Juli 2011, 20:58



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. *Anmerkung: Die ersten drei Klassen erinnern an das Entwurfsmuster „Composite“.*

Zeile	korrekter Programmtext
7	Mitspieler[] dieMitspieler = new Mitspieler[3];
11	Spiel dasSpiel = new Spiel(dieMitspieler, 2);
12	dasSpiel.SpieleBisZumEnde();
16	abstract class Zufallsgenerator
24	public override int Werfe()
27	return zufallsgenerator.Next() % 6 + 1;
46	Summe += w.Werfe();
52	enthalteneWuerfel.Add(new Wuerfel());
78	public Mitspieler(string name)
84	punktestand += punkte;
88	class Spiel
105	zufallsgenerator = new Wuerfelbecher(zahlDerWuerfel);
113	if (m.Punktestand >= 1000)
116	istBeendet = true;
123	while(!IstBeendet)

*Ebenfalls als korrekt gefundene Fehler bewertet:*

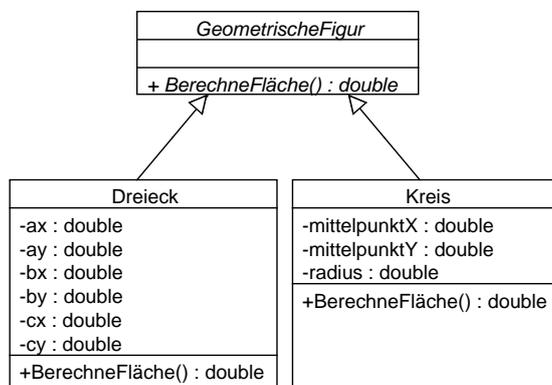
- *In Zeile 18 ist die Reihenfolge abstract public ebenso erlaubt wie public abstract. **Hauptsache, das int steht hinten.***
  - *Vor Zeile 93 sollte sinnvollerweise public stehen, auch wenn dieses Programm ohne das funktioniert.*
2. a ist "Egon", b ist 0 (*Es wird für diese Mitspieler keine Zufallsfunktion aufgerufen!*) und c ist "Doris".
  3. SpieleEineRunde sollte virtuell werden. Die Attribute sollten protected werden. Der Konstruktor könnte sein:

```
public KurzesSpiel(Mitspieler[] dieMitspieler)
: base(dieMitspieler)
{ }
```

4. Die Methode `NehmeEinenWurfelWeniger` wird eine **Exception** werfen, wenn sie aufgerufen wird, aber kein Würfel mehr vorhanden ist.

```
5. int sekundenAusHMS(string hms)
{
    string[] parts = hms.Split(':');
    int h = int.Parse(parts[0]);
    int m = int.Parse(parts[1]);
    int s = int.Parse(parts[2]);
    return (h * 60 + m) * 60 + s;
}
```

6. Zum Beispiel so:



7. Zum Beispiel so:

```
double strecke = 0.0;
for (int i = 0; i < messungen.Count-1; i++)
{
    double dauer = messungen[i + 1].Zeitpunkt
                    - messungen[i].Zeitpunkt;
    strecke += messungen[i].Geschwindigkeit * dauer;
}
```

*Mit anonymen Funktionen = Lambda-Ausdrücken klappt das ausnahmsweise nicht so gut.*

8. Zum Beispiel so:

```
public int Parse()
{
    return int.Parse(this);
}
```