

## Zwischenübung 7: Objektorientierte Programmierung

### Teste Dich selbst:

1. Was ist die Ausgabe folgenden Programms

```
class Test{
    public static void main(String[] args){
        int x = 5;
        foo(x);
        System.out.println(x);
    }
    static void foo(int x){
        x = 2;
    }
}
```

- a) Das Programm lässt sich nicht übersetzen
- b) Das Programm erzeugt einen Laufzeitfehler
- c) 5
- d) 2

2. Was ist die Ausgabe folgenden Programms

```
class Test{
    int x = 5;
    public static void main(String[] args){
        Test t = new Test();
        t.foo(t);
        System.out.println(t.x);
    }
    void foo(Test t){
        x = 2;
    }
}
```

- e) Das Programm lässt sich nicht übersetzen
- f) Das Programm erzeugt einen Laufzeitfehler
- g) 5
- h) 2

3. Was ist die Ausgabe folgenden Programms

```
class Test{
    int x = 5;
    public static void main(String[] args){
        Test t = new Test();
        t.foo(t);
        System.out.println(t.x);
    }
}
```

```
void foo(Test t){
    t.x = 2;
}
}
```

- i) Das Programm lässt sich nicht übersetzen
- j) Das Programm erzeugt einen Laufzeitfehler
- k) 5
- l) 2

4. Was ist die Ausgabe folgenden Programms

```
class Test{
    public static void main(String[] args){
        foo(1,2,3,4,5,6);
    }
    void foo(int... x){
        for(int i=0; i < x.length; i++){
            System.out.print(x[i]);
        }
    }
}
```

- m) Das Programm lässt sich nicht übersetzen
- n) Das Programm erzeugt einen Laufzeitfehler
- o) 123456
- p) 12345

5. Was ist die Ausgabe folgenden Programms

```
class Test{
    public static void main(String[] args){
        this.foo(1,2,3,4,5,6);
    }
    void foo(int... x){
        for(int i=0; i < x.length; i++){
            System.out.print(x[i]);
        }
    }
}
```

- q) Das Programm lässt sich nicht übersetzen
- r) Das Programm erzeugt einen Laufzeitfehler
- s) 123456
- t) 12345

6. Was ist die Ausgabe folgenden Programms

```
class Test{
    public static void main(String[] args){
        Test t = new Test();
        foo(1,2,3,4,5,6);
    }
    void foo(int... x){
        for(int i=0; i < x.length; i++){
            System.out.print(x[i]);
        }
    }
}
```

- u) Das Programm lässt sich nicht übersetzen
- v) Das Programm erzeugt einen Laufzeitfehler
- w) 123456
- x) 12345

7. Was ist die Ausgabe folgenden Programms

```
class Test{
    public static void main(String[] args){
        Test t = new Test();
        t.foo(1,2,3,4,5,6);
    }
    void foo(int... x){
        for(int i=0; i < x.length; i++){
            System.out.print(x[i]);
        }
    }
}
```

- y) Das Programm lässt sich nicht übersetzen
- z) Das Programm erzeugt einen Laufzeitfehler
- aa) 123456
- bb) 12345

## Übungsaufgabe (wie in Zwischenübung 6)

Implementieren Sie das beliebte Trinkspiel „Meier“ (auch „Mäxchen“ oder „Lügenmäxchen“) in Java. Die Spielregeln finden Sie bei Wikipedia: <http://de.wikipedia.org/wiki/Meiern>

- Die Anzahl der Spieler wird in der Kommandozeile beim Start des Programms angegeben.
- Es wird für jeden Spieler ein Objekt definiert und jeder Spieler kennt seinen Nachbarn (den er anlügen darf)
- In der Prozedur main wird außerdem ein Spieler aufgerufen, der beginnt. Dieser ruft dann seinen Nachbar auf und so weiter.
- Ist ein Spiel zu Ende, weil eine Lüge aufgedeckt wurde, wird dem folgenden Nachbar keine weitere Nachricht geschickt und das Programm terminiert.
- In der Eingabeaufforderung soll dabei immer zu sehen sein, wer was gewürfelt hat und was er dann sagt.