

NAME DER DOZENTEN: MB,JH,KH,CK,DV,FZ



Portfolioprüfung (Artefakt, Beispiel): Einführung in die OO-Programmierung

QUARTAL: III.

Name des Prüflings:

Matrikelnummer:

Zenturie:

Dauer: 30 Minuten

Anzahl Seiten **ohne** Deckblatt: 5

Datum:

Hilfsmittel: keine.

Bemerkungen:

- Bitte prüfen Sie zunächst die Teilprüfungsleistung auf Vollständigkeit.
- Bitte lösen Sie nicht die Heftung.

Es sind 30 Punkte erreichbar.

Aufgabe	Erreichbare Punkte	Erreichte Punkte
1	3	
2	5	
3	5	
4	5	
5	5	
6	5	
7	2	
Summe	30	

Datum: _____

Unterschrift: _____

1. (3 Punkte) Der Java Compiler ...

- ☐ überprüft zur Laufzeit die sichere Ausführung der Anweisungen.
- ☐ überprüft die Syntax von Java Programmen und zeigt Fehler an.
- ☐ übersetzt Java Sourcecode in maschinenunabhängigen Bytecode.
- ☐ trägt den namen javac und ist Teil des JDK. javac wird beim Download des JDK mit ausgeliefert.
- ☐ ist nur zum Auffinden von Fehlern erforderlich. Ein JIT-Compiler kann den Sourcecode direkt ohne den Zwischenschritt über den Bytecode interpretieren.

2. (5 Punkte) Gegeben sei die folgende Klasse:

```
public class Point {  
    private final int x;  
    private final int y;  
    public Point(int xPos, int yPos){  
        x=xPos;  
        y=yPos;  
    }  
}
```

Schreiben Sie für die Klasse `Point` eine Methode `isOnXAxis`, die feststellt, ob ein Punkt auf der X-Achse liegt. Den o. g. Code brauchen Sie nicht abzuschreiben.

[illegible]

3. (5 Punkte) Welche der folgenden Schleifen sind korrekte Schleifen in Java und keine Endlosschleifen:

- ☐

```
int i, j;  
for (i = 1, j = 10; i <= 10; j++) {  
    System.out.println(i+" "+j);  
}
```
- ☐

```
while (int i>0) {  
    i=i-1;  
}
```
- ☐

```
int i = 0;  
do {  
    i++;  
    System.out.println(" Jetzt zum "+i+".Mal!");  
} while(i <= 10);
```
- ☐

```
int number=10;  
int sum= 0;  
repeat {  
    sum = sum + number;  
    number = number - 2;  
} until number == 0
```
- ☐

```
int i=5;  
while (i > 0) {  
    System.out.println(" Countdown: "+i);  
    i=i-1;  
}
```
- ☐

```
for i in range(1,100) {  
    sum = sum + i;  
}  
System.out.println(sum);
```
- ☐

```
for (;;);
```

4. (5 Punkte) Schreiben Sie eine Methode `fill`, die ein Ergebnis-Array, mit einer übergebenen Größe $N > 0$, mit der jeweiligen Potenz i^2 der Position im Array befüllt.

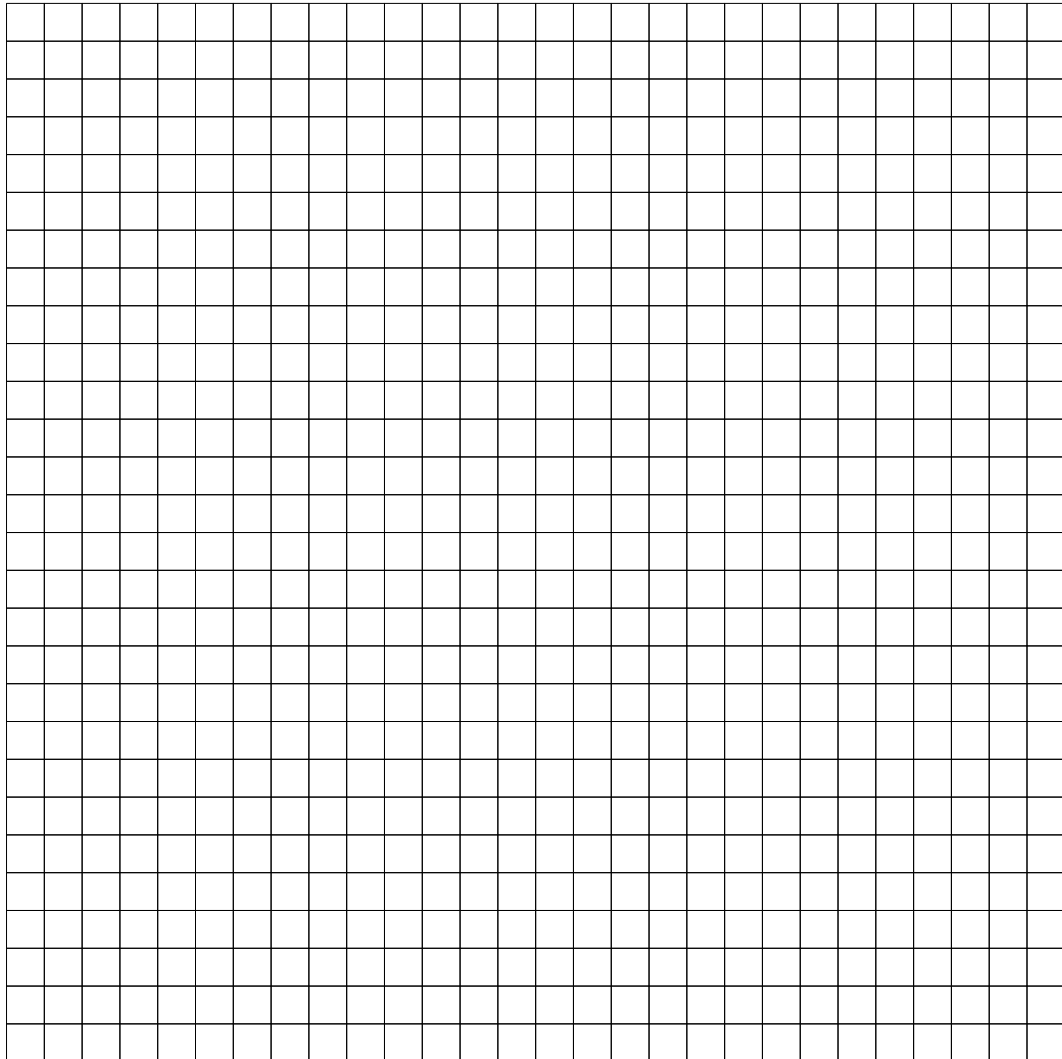
Beispiel

`a[0]` ist 0

`a[1]` ist 1

`a[2]` ist 4

`a[3]` ist 9



-
- A full-page sheet of white graph paper featuring a uniform grid of thin black lines. The grid consists of small squares covering the entire area. There are no margins, text, or other markings on the page.

6. (5 Punkte) Schreiben Sie eine Methode `printArrayElements`, die ein `String-Array` als Parameter erhält. Implementieren Sie die Methode so, dass sie mit Hilfe einer `for-each-Schleife` über das übergebene Array iteriert und nur jene Element, die einen `String` enthalten, in jeweils einer Zeile ausgibt. Das Array kann `null-Referenzen` als Elemente enthalten.

A full-page sheet of white graph paper with a light gray grid. The grid consists of small squares, approximately 10 units wide by 10 units high. There are no margins or additional markings on the page.

7. (2 Punkte) Was sind valide Methodensignaturen? Markieren Sie diese.
- A. **public void** methodName ()
 - B. **public int** methodName ()
 - C. **public void** methodName () {}
 - D. **public void** methodName
 - E. **public long[]** methodName (**int** parameter)
 - F. **public** String methodName (**int** parameter1, String parameter2)
 - G. **public void** methodName (parameter1: **int**; parameter2: String)