

Variablenzugriff (5 Punkte)

Eine Klasse A enthalte die folgende Variablendeklaration:

```
static int x;
```

Welche der folgenden Aussagen treffen zu?

- A ☒ Auf einem Objekt a mit statischem Typ A ist der folgende Zugriff erlaubt: a.x;
- B ☒ Es sei B eine von A abgeleitete Klasse in einem anderen Paket als A und b ein Objekt mit statischem Typ B. Dann ist der folgende Zugriff erlaubt: b.x;
- C ☒ Der Zugriff A.x ist erlaubt.
- D ☐ Es sei nun C von A abgeleitet und im gleichen Paket wie A. Weiterhin enthalte C die folgende Variablendeklaration:
- ```
static float x;
```
- Gegeben sei weiter A c = new C();
- Wegen dynamischer Bindung wird dann bei dem Aufruf c.x die in C definierte Variable x zurückgegeben.
- E ☐ Es sei nun C von A abgeleitet und im gleichen Paket wie A. Weiterhin enthalte C die folgende Variablendeklaration:
- ```
static float x;
```
- Dann kompiliert das Programm nicht.

A getestet
B nur im selben Paket getestet falsch
C getestet
D getestet
E getestet, vermute, es fehlt etwas in der Angabe

Dann kompiliert das Programm nicht.

Objektorientierung und Polymorphie (5 Punkte)

?? keine Ahnung ??

Welche der folgenden Aussagen über Objektorientierung und Polymorphie sind wahr?

- A ☒ Der Cast Operator ist reflexiv, das heißt man kann jeden Typ A auf sich selbst casten.
- B ☐ Der Cast Operator ist symmetrisch, das heißt wenn man A auf B casten kann, kann man auch B auf A casten.
- C ☒ Der Cast Operator ist transitiv, das heißt wenn man A auf B casten kann und B auf C casten kann, kann man auch A auf C casten.
- D ☒ Der Cast Operator ist für Klassen antisymmetrisch, das heißt für alle Paare von Klassen A und B gilt: wenn man A auf B casten kann und B auf A casten kann, dann gilt A = B
- E ☐ Es seien B und C zwei beliebige Klassen, dann gibt es immer eine Klasse A so dass die folgenden beiden Aussagen wahr sind:
- (1) A ist eine Oberklasse von B oder A = B
- (2) A ist eine Oberklasse von C oder A = C

Zeichenketten (5 Punkte)

Welche der folgenden Aussagen zu Zeichenketten (Strings) in Java sind korrekt?

- A ☒ "Kurs 1622" ist ein String Literal.
- B ☐ Es ist möglich, dass a == b true zurückgibt, obwohl die Zeichenketten in a und b verschieden sind.
- C ☒ Es ist möglich, dass a und b die gleichen Zeichenketten enthalten, aber a == b false zurückgibt.
- D ☐ Strings gehören in Java zu den primitiven Datentypen.
- E ☐ Ein String kann maximal 32.767 Zeichen enthalten.

String object:
String s = new String("xx");
String s = "xx"

String literal:
"xx"

Exceptions (5 Punkte)

Welche der folgenden Aussagen über Exceptions sind wahr?

- A ☐ Wenn eine Exception auftritt, wird die Programmausführung abgebrochen.
- B ☒ Es ist möglich, dass eine Methode keine throws Anweisung enthält und dennoch eine Exception wirft.
- C ☒ Die Ausführung der folgenden Programmzeilen erzeugt eine Exception:

```
int[] x = new int[5];  
x[5]=2;
```
- D ☒ Die Ausführung der folgenden Programmzeile erzeugt eine Exception:

```
int x = 5/0;
```
- E ☐ Die Ausführung der folgenden Programmzeile erzeugt eine Exception:

```
for(int i=1; i>0; ++i) System.out.println(i);
```

A Exception kann gefangen werden ?
B Methode wirft Exception ohne throws: stack-overflow ?
C out-of-bounds
D division durch 0
E läuft bis Integer.MAXValue bricht ab bei MINValue

Dynamische Programmierung (4 Punkte)

Welche der folgenden Aussagen zur dynamischen Programmierung sind korrekt?

- A ☒ Memoisation führt bei jeder rekursiven Lösung zu einer Verbesserung der Laufzeit.
- B ☐ Die optimale Teilstruktureigenschaft ist notwendig, damit im ersten Schritt der dynamische Programmierung, der Charakterisierung der Struktur einer optimalen Lösung, eine korrekte Lösung des Optimierungsproblems gefunden werden kann.
- C ☒ Zum Beweis der Eigenschaft der überlappenden Teilprobleme ist es hinreichend, ein einzelnes Beispiel anzugeben, bei dessen rekursiver Lösung zwei Mal die gleiche Unterprobleminstanz auftritt.
- D ☐ Die Memoisation kann zur Beschleunigung einer rekursiven Lösung eingesetzt werden, ohne irgendwelche Nachteile mit sich zu bringen.

A falsch (Fakultät)
B vermutlich falsch ?
C
D geringfügig längere Laufzeit und mehr Speicher

Zähle alle Quadratzahlen für die ganzen Zahlen bis 100.000.

Quadratzahl = Zahl, die das Quadrat einer anderen Zahl ist.

Exponentialreihe

e^x kann mit folgender Formel angenähert werden:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

Implementieren Sie eine iterative Methode, die e^x mit Hilfe der ersten k Summanden annähern soll:

```
public double exp(double x, int k)
```

```
    // ...
```

```

boolean stringcheck(string x){
    byte s = 0;
    //Durchlaufe die Zeichenkette x vom ersten bis zum letzten Zeichen
    for(int i = 0; i < x.length(); ++i)
        switch(s){
            case 0:
                s=1;
                break;
            case 1:
                s=0;
                break;
        }
    // Gib true zurück, falls s=0 ist.
    return s==0;
}

```

- A ☒ Die Methode gibt true für die leere Zeichenkette zurück.
- B ☐ Die Methode überprüft die Zeichenkette darauf ob sie kleingeschrieben ist.
- C ☒ Die Methode überprüft die Zeichenkette darauf, ob sie eine gerade Länge hat.
- D ☐ Die Methode gibt true zurück genau dann wenn die Zeichenkette leer ist.
- E ☐ Die Methode wirft für keine Eingabe eine Exception.
- F ☐ Die switch Anweisung ist fehlerhaft, weil es keinen default Fall gibt.
- G ☐ Es ist möglich, die Methode mit einem beliebigen Objekt einer beliebigen Klasse aufzurufen.
- H ☐ Für alle Eingaben würde die selbe Ausgabe herauskommen, wenn man die Anweisung ++i in der for Schleife durch i++ ersetzen würde.
- I ☐ Für alle Eingaben würde die selbe Ausgabe herauskommen, wenn man die Anweisung i=0 in der for Schleife durch i=1 ersetzen würde und die Anweisung

```

byte s = 0;
durch
byte s=1;
ersetzen würde.

```

A getestet
 B getestet
 C getestet
 D getestet
 E ausgenommen man übergibt keinen falschen Datentyp
 F kein Fehler in Eclipse
 G nicht public
 H falsch, verlesen
 I getestet, leerer String unterscheidet sich

Die Methode boolcheck (6 Punkte)

Gegeben sei die folgende Methode boolcheck. Welche Aussagen über boolcheck sind korrekt?

```

public static boolean boolcheck(boolean x, y, z){
    // Überprüfe ob die booleans x, y und z zusammenpassen.
    return x && y && !z || !x && y && z || x && !y && z;
}

```

- A ☐ Wenn wir alle && durch & ersetzen, gibt es Eingaben, für die eine andere Ausgabe heraus kommt.
- B ☐ Es gibt keine Eingabe, für die die Methode boolcheck eine Exception wirft.
- C ☐ Die Methode liefert true zurück genau dann wenn x und y true sind.
- D ☒ Die Methode liefert true zurück, genau dann wenn zwei der Eingaben true sind.
- E ☒ Die Ausgabe der Methode ist unabhängig von der Reihenfolge der Parameter, die übergeben werden.
- F ☐ Die Methode gibt true zurück genau dann wenn x und y den gleichen Wert haben.

A getestet
 B außer man übergibt anderen Datentyp
 C getestet
 D getestet
 E solange 2 true sind, verstehe Frage nicht ganz
 F getestet

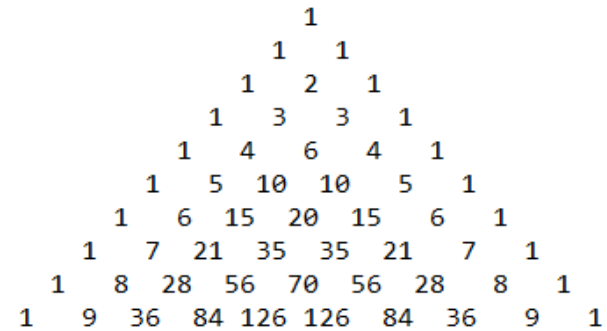
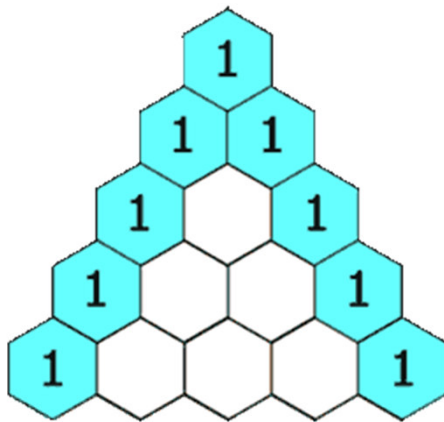
Die Methode intcheck (10 Punkte)

Gegeben sei die folgende Methode intcheck. Welche Aussagen über intcheck sind korrekt?

```
public static boolean intcheck(int x){
    if(x<=1) return false;
    //Durchlaufe alle Zahlen von 2 bis x/2 und prüfe ob i einen Widerspruch zeigt.
    for(int i=2; i<=x/2; ++i);
        if(x/i!=x) return false;
    return true;
}
```

- A ☒ Die Methode liefert true für 17 zurück.
- B ☐ Die Methode liefert false für 11 zurück.
- C ☒ Die Methode liefert false für 4 zurück.
- D ☒ Die Methode überprüft ob die Eingabe ungerade ist.
- E ☒ Die Methode überprüft ob die Eingabe prim ist.
- F ☒ Es ist unmöglich dass in dieser Methode ein Variablenüberlauf einer int Variable geschieht.
- G ☐ Für die Eingabe 50 wird true ausgegeben.
- H ☐ Die Methode ist rekursiv.
- I ☐ Die Überprüfung ob $x/i \neq x$ ist, ist redundant, da sich die beiden i kürzen
- J ☒ Wenn wir den Ausdruck $x/i \neq x$ durch den Ausdruck $x \cdot i / i$ ersetzen, ändert sich die Ausgabe bei einigen Eingaben.

A getestet
B getestet
C getestet
D **falsch**, vermutlich verklickt
E nur wenn Zahl natürlich teilbar ist kann $\text{int } x / \text{int } i$ eine Ganzzahl ergeben, die $*i == x$
F wenn Integer.MAX_VALUE übergeben wird, wird i max MAX_VALUE / 2, $(\text{int})(x/i * i) <= x$
G getestet
H for Schleife, kein weiterer Aufruf
I $\text{int } (x/i) * i \neq \text{double}(x/i)$
J getestet (alle $x > 2$ werden false)



Programmiere die Ausgabe eines Pascal'schen Dreiecks für 10 Zeilen. Jedes Element ist die Summe der beiden darüber liegenden Elemente.

$\text{coef} * (i - k + 1) / k$

i = Zeile, k = Spalte