



Informationen zur Klausur

**DIE KLAUSUR LIEGEN LASSEN,
BIS SIE LOSLEGEN DÜRFEN!**

Prüfungsdauer: 90 Minuten

Wertung: Insgesamt können 100 Punkte erreicht werden. Die pro Aufgabe bzw. Teilaufgabe erzielbaren Punkte sind angegeben. *Die erreichten Punkte werden Ihnen hälftig auf die Gesamtleistung angerechnet.*

Hilfsmittel: Es sind keine Hilfsmittel erlaubt. Es darf kein eigenes Papier verwendet werden.

Hinweise:

- Bitte verwenden Sie keinen Rotstift oder Bleistift.
- Ihre Lösungen tragen Sie bitte nur auf dem Lösungsblatt ein.
- Schreiben Sie Ihre Antworten bitte ausschließlich in die auf dem Lösungsblatt vorgesehenen Abschnitte.
- Sie müssen mit den vorhandenen leeren Blättern für Ihre Notizen und Lösungsskizzen auskommen. Sie bekommen kein weiteres Papier.
- Es werden nur leserliche Klausurlösungen bewertet!
- Die Angabe der Lösungen hat sich in Art und Form an den in der Veranstaltung verwendeten Konventionen zu orientieren.
- Sie geben am Ende der Klausur nur das Lösungsblatt ab. Die Aufgabenblätter sind für den Verbleib bei Ihnen bestimmt.



Liebe Studierende,

zu Beginn ein paar formale Infos rund um die Klausur:

- Tragen Sie Ihren Namen und Ihre Matrikelnummer auf dem Lösungsblatt ein und unterschreiben Sie das Lösungsblatt.
- Während der Klausur werden keine Fragen zum Verständnis oder zur Klärung einer Klausuraufgabe beantwortet.
- Smartphones, Smartwatches, Kopfhörer und andere Hightech-Geräte dürfen aus verständlichen Gründen nicht verwendet werden.
- Sie verlassen den Platz nur mit unserer Erlaubnis. Melden Sie sich, wenn etwas ist.
- Das gilt auch, wenn Sie vorzeitig die Klausur abgeben wollen: Melden Sie sich. Wir sammeln das Lösungsblatt ein. Verlassen Sie anschließend bitte leise den Saal.
- 20 Minuten vor Ende der Klausur bitte keine vorzeitigen Abgaben mehr. Das macht zu viel Unruhe und erzeugt Stress.
- Legen Sie Ihren Studierendenausweis an den äußeren Rand ihres Tisches.
- In die Lösungsfelder bitte erst dann eine Lösung eintragen, wenn Sie sich sicher sind.

Viel Erfolg und herzliche Grüße,

Dominikus Herzberg

Aufgabenteil zur PiS-Klausur, SoSe 2022

Prüfer: Prof. Dr. Dominikus Herzberg, Fachbereich MNI



Bitte nutzen Sie für Ihre Antworten ausschließlich die separat ausgehändigten Lösungsblätter! Sie geben am Ende der Klausur nur die Lösungsblätter ab! Nutzen Sie die Rückseiten der Aufgabenblätter als Notizblätter.

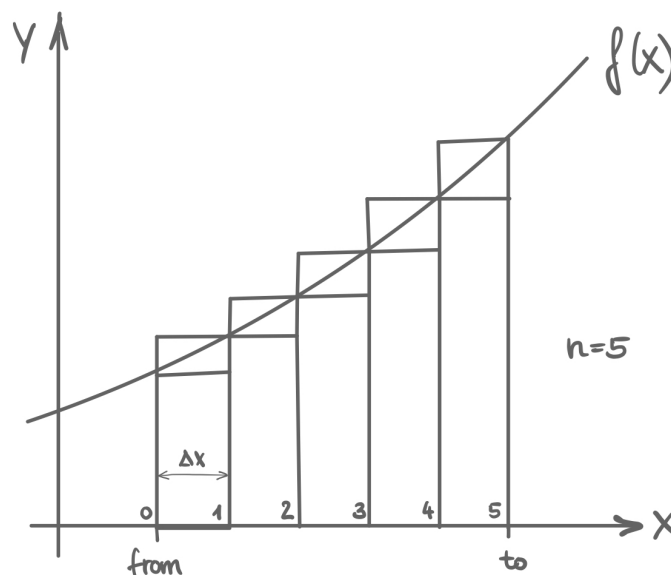
1 Objektvergleich (12 + 4 = 16 Punkte)

Gegeben sei eine Klasse `Ellipse`, die die Achsen einer Ellipsen-Instanz mittels zweier privater `double`-Variablen `height` und `width` verwaltet.

1. Implementieren Sie die Methode `equals` nach dem in der Veranstaltung erlernten Schema.
2. Implementieren Sie die Methode `int hashCode` für `Ellipse` in einer Weise, so dass kein besonderes Verständnis zur Berechnung eines HashCodes erforderlich ist.

2 Integrieren (28 Punkte)

Mit Hilfe des Integrals bestimmt man in der Mathematik die Fläche unter einer Kurve, die durch eine Funktion $f(x)$ beschrieben ist. Dafür gibt es ein einfaches Näherungsverfahren, das in dem Bild dargestellt ist: Der Bereich, innerhalb dessen die Fläche bestimmt werden soll (von *from* bis *to*), wird in n Abschnitte unterteilt; im Beispiel ist $n = 5$, es gilt $\Delta x = (to - from)/n$. Nun werden die Flächen der Rechtecke berechnet, die „unterhalb“ der Kurve liegen, und die Flächen der Rechtecke, die „oberhalb“ der Kurve liegen. Die Flächen werden aufaddiert und durch 2 geteilt, was eine Näherung für den Flächeninhalt darstellt.



Der eine Satz von Rechtecken (*rectangles*) „unterhalb“ der Kurve hat die Höhen, die durch die Funktionswerte an den von 0 bis 4 ausgewiesenen Stellen bestimmt sind; der andere Satz von Rechtecken „oberhalb“ der Kurve hat die Höhen an den von 1 bis 5 ausgewiesenen Stellen.

Gegeben sei der folgende Record Rect. Das neue Java-Sprachfeature eines Records legt, vereinfacht gesprochen, eine (Daten)Klasse an, deren Konstruktor den Angaben im Record entspricht. Eine Instanz wird wie gewohnt mit new erzeugt, die übergebenen Werte werden automatisch in den entsprechenden Variablen gespeichert. area ist eine ganz normale Methode zu dieser Datenklasse. xPos und yPos sind die Koordinaten der linken unteren Ecke eines Rechtecks.

```
record Rect(double xPos, double yPos, double height, double width) {  
    double area() { return height * width; }  
}
```

Schreiben Sie für die JShell eine Methode namens integrate mit dem gegebenen Parameter- und Codegerüst; die an f übergebene Funktion soll als Lambda-Ausdruck formulierbar sein.

```
double integrate(          f,          from,          to,          n) {  
    assert  
  
    double areaR =  
  
    double areaL =  
  
    return  
}
```

Die Flächen areaR und areaL sollen in den folgenden Schritten berechnet werden:

1. Sie beginnen mit einem IntStream für die Werte von 0 bis $n - 1$ für areaR bzw. 1 bis n für areaL.
2. Im nächsten Schritt bestimmen Sie die entsprechenden x-Werte
3. Es werden die Rechtecke erzeugt
4. Anschließend wird die Fläche der Rechtecke bestimmt
5. Aus dem Strom der Flächenwerte wird die Gesamtfläche bestimmt

3 Die Kreiszahl Pi (π) schätzen (3 + 3 + 9 = 15 Punkte)

Es gibt ein sehr einfaches Verfahren, die Kreiszahl $\pi \approx 3,1415926 \dots$ zu schätzen: Auf einem quadratischen Papier mit lauter zufällig gesetzten Punkten zeichne man einen maximal großen Kreis ein. Die Anzahl der Punkte im Kreis im Verhältnis zur Gesamtzahl aller Punkte ergibt einen Näherungswert für π . Das bilden wir als Programm für die JShell ab.

1. Ein Punkt mit den Koordinaten (x, y) hat vom Ursprung, dem Punkt mit den Koordinaten $(0, 0)$, einen Abstand von $d = \sqrt{x^2 + y^2}$. Ein Punkt liegt im Einheitskreis, wenn $d \leq 1$ ist. Erstelle eine Variable isInCircle vom Typ BiPredicate. Der Variable ist ein Ausdruck zugewiesen, der ermittelt, ob ein durch x und y gegebener Punkt innerhalb des Einheitskreises liegt.
2. Legen Sie einen Supplier namens hit für boolsche Werte an, der den boolschen Wert mit Hilfe von isInCircle und Zufallswerten zwischen 0 und 1 erzeugt. Der Aufruf Math.random() generiert eine double-Zahl von 0 bis 1.
3. Schreiben Sie eine Methode double estimatePi(long n), wobei n die Anzahl der Anwendungen von hit entspricht. Im Rumpf der Methode bekommt die Variable long circleHits die Anzahl der Kreistreffer zugewiesen; dazu ist ein Strom von hits zu generieren, wovon die hits, die true sind, ausgezählt werden. Übrigens ist der

Verhältniswert zu vervierfachen, da die Zufallszahlen nur Werte in einem Viertelstück des „Papiers“ generieren.

4 Collections (4 + 6 + 3 + 2 = 15 Punkte)

1. Erstellen Sie eine Variable `listOfSetOfNumbers`, die eine immutable Liste mit immutablen Mengen von Ganzzahlen verwaltet. Die Variable werde zugleich initialisiert, wobei die erste Menge aus den Zahlen 2, 5 und 6 besteht und die zweite Menge aus den Zahlen 1, 4, 5 und 8.
2. Erstellen Sie eine klassenfreie Methode (so wie man das in der JShell ohne Klassenkontext machen kann) namens `level`, die eine Menge von Ganzzahlen entgegen nimmt und eine *neue* Menge von Ganzzahlen zurückliefert. Die Methode halbiert gerade Zahlen und verdoppelt ungerade Zahlen. Für die Implementierung wird nur ein Semikolon benötigt.
3. Erstellen Sie einen Ausdruck, der aus `listOfSetOfNumbers` eine *neue* Liste erstellt, wobei die *neuen* Mengen in der Liste aus der Anwendung von `level` hervorgehen.
4. Was liefert der Ausdruck als Ergebnis, wenn er in der JShell ausgeführt wird?

5 Map (4 + 4 = 8 Punkte)

1. Erstellen Sie eine Variable namens `calc`, der eine Map zugewiesen wird, wobei den Zeichen `+` und `-` jeweils eine Funktion in Form eines Lambda-Ausdrucks zugeordnet wird; in dem einen Fall addiert die Funktion zwei Ganzzahlen, im anderen subtrahiert sie zwei Ganzzahlen.
2. Erstellen Sie einen Ausdruck, der die Variable `calc` nutzt, um auf die mit dem Additionszeichen assoziierte Funktion zuzugreifen, und die Funktion mit den Werten 2 und 3 aufruft.

6 Fragen (6 × 3 = 18 Punkte)

Hinweis: Pro Frage kann mehr als eine Antwort korrekt sein. Die Fragen sind so formuliert, dass in der Regel nicht ableitbar ist, ob eine oder mehr Antworten zutreffen.

Frage A: Die JVM ist als nebenläufige Architektur aufgesetzt. Welche Ressource verwaltet ein Thread separat für sich, d.h. die Ressource wird nicht mit anderen Threads geteilt?

1. Program Counter
2. Nativer Methodenstapel
3. Run-Time Constant Pool
4. JVM-Stapel
5. Heap

Frage B: Ein Thread kann sich in dem folgenden Zustand befinden:

1. COMPLETED
2. COMPETING
3. BLOCKED
4. TERMINATED
5. RECYCLED

Frage C: Welche alternativen Möglichkeiten der nebenläufigen und parallelen Programmierung gibt es, statt direkt mit Threads zu arbeiten?

1. Reaktive Programmierung
2. Logische Programmierung
3. Strombasierte Programmierung
4. Ergebnisbasierte Programmierung
5. Nachrichtenbasierte Programmierung

Frage D: Welche Syntax für einen Lambda-Ausdruck ist gültig?

1. `(x) -> x + 1`
2. `() -> {}`
3. `(String s) -> s.length()`
4. `int x -> x++`
5. `t -> { t.start(); }`

Frage E: Eine Klasse E erweitere die Klasse X. Was syntaktisch möglich ist, ist semantisch zu überprüfen. Welche Aussage diesbezüglich gültig?

1. Die erweiterte Klasse muss den *isa*-Test bestehen
2. Die erweiterte Klasse muss isomorph zur Oberklasse sein
3. Die Klassenhierarchie sollte nicht breiter sein als sie tief ist
4. Die Klassen müssen das Liskov'sche Substitutionsprinzip erfüllen
5. Klassennamen sind wichtig, da der Name die Syntax bestimmt

Frage F: Welche Aussage zu Datenstrukturen ist korrekt?

1. Eine Liste ist eine assoziative Verknüpfung von Listenelementen
2. Eine Map ist von der Zugriffszeit genauso effizient wie eine Liste
3. Eine Queue ist dasselbe wie eine Liste
4. Ein Set kann nicht leer sein
5. Keine der vorigen Antworten trifft zu

(Notizblatt)

(Notizblatt)



Lösungsblatt zur PiS-Klausur

Sie geben am Ende der Klausur nur das Lösungsblatt ab!

Nachname, Vorname	Matrikelnummer	Note
Unterschrift 22.09.2022		

1. Objektvergleich (12 + 4 = 16)

1.

2.

2. Integrieren (28)

double integrate(



3. Kreiszahl Pi schätzen ($3 + 3 + 9 = 15$)

- 1.
- 2.
3. `double estimatePi(long n) {`

4. Collections ($4 + 6 + 3 + 2 = 15$)

- 1.
- 2.
- 3.
- 4.

5. Map ($4 + 4 = 8$)

- 1.
- 2.

6. Fragen ($6 \times 3 = 18$)

1. 2. 3. 4. 5.	Punkte	1. 2. 3. 4. 5.	Punkte
A <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>		D <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	
B <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>		E <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	
C <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>		F <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	

Sollten Sie eine Lösung revidieren wollen, streichen Sie die betreffende Reihe durch und notieren Sie hier neben der Tabelle Ihre korrigierte Lösung im Stil von Z13 (die hypothetische Frage Z, Antwort 1 und 3).