

PD Dr. rer. nat. Robert Heinrich Institute of Information Security and Dependability (KASTEL) https://sdq.kastel.kit.edu/wiki/Programmieren/ programmieren-vorlesung@cs.kit.edu

Programmieren - Wintersemester 2023/24

	5 Bonuspunkte	Ausgabe: Abgabe: Abgabefrist:	25.10.2023, ca. 12:00 Uhr 02.11.2023, 12:00 Uhr 10.11.2023, 06:00 Uhr
--	---------------	-------------------------------------	---

Geschlechtergerechte Sprache

Wenn das generische Maskulinum gewählt wurde, geschieht dies zur besseren Lesbarkeit und zum einfachen Verständnis der Aufgabenstellung. Sofern nicht anders angegeben, beziehen sich Angaben im Sinne der Gleichbehandlung auf Vertretende aller Geschlechter.

Plagiarismus

Es werden nur selbstständig angefertigte Lösungen akzeptiert. Das Einreichen fremder Lösungen, seien es auch nur teilweise Lösungen von Dritten, aus Büchern, dem Internet oder anderen Quellen, ist ein Täuschungsversuch und führt zur Bewertung "nicht bestanden". Ausdrücklich ausgenommen hiervon sind Quelltextschnipsel von den Vorlesungsfolien und aus den Lösungsvorschlägen des Übungsbetriebes in diesem Semester. Alle benutzten Hilfsmittel müssen vollständig und genau angegeben werden. Alles, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde, muss deutlich kenntlich gemacht werden.

Studierende, die den ordnungsgemäßen Ablauf einer Erfolgskontrolle stören, können von der Erbringung der Erfolgskontrolle ausgeschlossen werden. Ebenso stellt unter anderem die Weitergabe von Teilen von Testfällen oder Lösungen bereits eine Störung des ordnungsgemäßen Ablaufs dar. Auch diese Art von Störungen können ausdrücklich zum Ausschluss der Erfolgskontrolle führen.

Kommunikation und aktuelle Informationen

In unseren $FAQs^1$ finden Sie einen Überblick über häufig gestellte Fragen und die entsprechenden Antworten zum Modul "Programmieren". Bitte lesen Sie diese sorgfältig durch, noch bevor Sie Fragen stellen, und überprüfen Sie diese regelmäßig und eigenverantwortlich auf Änderungen.

¹https://sdq.kastel.kit.edu/wiki/Programmieren/FAQ

In den *ILIAS-Foren* und auf *Artemis* veröffentlichen wir gelegentlich wichtige Neuigkeiten. Eventuelle Korrekturen von Aufgabenstellungen werden ebenso auf diesem Weg bekannt gemacht. Das aktive Beobachten der Foren wird daher vorausgesetzt.

Überprüfen Sie das Postfach Ihrer *KIT-Mailadresse* regelmäßig auf neue E-Mails. Sie erhalten unter anderem eine Zusammenfassung der Korrektur per E-Mail an diese Adresse. Alle Anmerkungen können Sie anschließend im Online-Einreichungssystem² einsehen.

Abgabehinweise

Die Abgabe im Online-Einreichungssystem wird am 02.11.2023, 12:00 Uhr, freigeschaltet. Achten Sie unbedingt darauf, Ihre Dateien im Einreichungssystem bei der richtigen Aufgabe vor Ablauf der Abgabefrist am 10.11.2023, 06:00 Uhr, hochzuladen. Beginnen Sie frühzeitig mit dem Einreichen, um Ihre Lösung dahingehend zu testen, und verwenden Sie das Forum, um eventuelle Unklarheiten zu klären. Falls Sie mit Git abgeben, *muss immer* auf den main-Branch gepusht werden.

• Geben Sie online Ihre *. java-Datei zur Aufgabe C in Einzelarbeit mit der entsprechenden Ordnerstruktur im zugehörigen Verzeichnis ab.

²https://artemis.praktomat.cs.kit.edu/

Aufgabe A: Vorbereitung IDE & Java (0 Punkte)

Für die Bearbeitung der Übungen können Sie eine Entwicklungsumgebung (IDE) Ihrer Wahl verwenden. Nachfolgend finden Sie eine Anleitung zur Einrichtung von Eclipse, bei anderen IDEs (z.B. IntelliJ oder Visual Studio Code) können einzelne Schritte von der Anleitung abweichen.

 Installieren Sie Java 17 (64 bit) auf Ihrem Gerät. Sie finden beispielsweise Open JDK unter https://jdk.java.net/archive/. Für Linux nutzen Sie am besten den Paketmanager Ihrer Distribution. Installationsanweisungen können Sie einer beliebigen Anleitung für Ihr Betriebssystem entnehmen. Im Folgenden einige Beispiele:

```
Ubuntu - https://techviewleo.com/install-java-openjdk-on-ubuntu-linux/
```

Windows-https://java.tutorials24x7.com/blog/how-to-install-openjdk-17-on-windows

Mac-https://knasmueller.net/how-to-install-java-openjdk-17-on-macos-big-sur

- 2. Laden Sie sich die aktuelle Version von Eclipse herunter: https://www.eclipse.org/downloads/. Achten Sie darauf, die richtige Eclipse-Version herunterzuladen oder im Installations-Wizard auszuwählen (Eclipse IDE for Java Developers).
- 3. Starten Sie Eclipse. Beim ersten Start werden Sie aufgefordert einen eclipse-workspace festzulegen. Sie können hier die Standardeinstellung übernehmen oder einen anderen Speicherort auswählen.
- 4. In diesem Übungsblatt werden wir mit einem fertigem Projekt, dem sogenannten Hamster-Simulator arbeiten. Laden Sie sich dazu aus ILIAS die Datei exercise-sheet-0.zip herunter und entpacken Sie diese. Importieren Sie das Projekt nun in Eclipse, indem Sie die folgenden Schritte befolgen: File → Import..., wählen sie hier Maven aus der Liste aus und dann Existing Maven Projects. Navigieren Sie zum Speicherort des entpackten exercise-sheet-0 Ordner und wählen Sie diesen aus. Es sollte nun eine /exercise-sheet-0/pom.xml Auswahl geben. Stellen Sie sicher, dass diese ausgewählt ist und klicken Sie auf Finish. Nun können Sie mit Aufgabe B fortfahren.
- 5. Hinweis: In diesem Übungsblatt werden Sie mit einem bereits voreingerichtetem Maven-Projekt arbeiten. Falls Sie Ecplise nicht mit dem Installer installiert haben, müssen sie noch das Plugin M2E manuell installieren, um Maven auch in Eclipse nutzen zu können. Gehen Sie dafür wie folgt vor: Help → Install New Software, fügen Sie im Textfeld Work with die URL https://download.eclipse.org/technology/m2e/releases/latest/ ein. Öffnen Sie das Untermenü Maven Integration for Eclipse und wählen Sie in der Liste das Plugin m2e Maven integration for Eclipse (includes...) aus. Klicken Sie dann auf Next > und folgen Sie den weiteren Anweisungen.

Aufgabe B: Erste Schritte mit dem Hamster

(0 Punkte)

B.1 Einführung in den Hamster-Simulator (0 Punkte)

In dieser Aufgabe lernen Sie eine Mini Programming World, den Hamster-Simulator, kennen. Im Hamster Simulator können Sie den kleinen verfressenen Hamster Paule steuern. Während Sie den Hamster in dieser Aufgabe noch mit einer graphischen Oberfläche (UI) steuern, werden Sie in Aufgabe C Paule programmatisch steuern. Sie sollten ggf. zwischen den Teilaufgaben das Programm neustarten, um das Spielfeld auf den Ausgangszustand zurückzusetzen.

- 1. Sie sollten in ihrer IDE im Projektverzeichnis nun das Projekt exercise-sheet-0 sehen. Wählen Sie dieses durch Doppelklick aus. Navigieren Sie zu src/main/java → edu.kit.informatik.
- 2. Wählen Sie die Datei FirstHamsterGameApp.java. Starten Sie das Programm in Ihrer IDE. Sie können das Programm in Eclipse über den grünen Run-Knopf kompilieren und ausführen. Bei erfolgreicher Ausführung sollte sich ein zweites Fenster mit einer UI für das Spielfeld öffnen.
- 3. Machen Sie sich mit der UI vertraut. Links befindet sich das Spielfeld (territory), auf der sich der Hamster (paule), sowie einige Körner (grains) befinden. Rechts davon befinden sich das Log und die UI Elemente um Klassen und Objekte zu inspizieren. Bspw. können Sie über die UI Elemente rechts Paules Fähigkeiten (*Methoden*) aufrufen. Dadurch können Sie Paule steuern. Diese Methoden können Sie später verwenden, um as Spielgeschehen programmatisch zu kontrollieren.
- 4. Verschaffen Sie sich einen groben Überblick über die Fähigkeiten Paules.
- 5. Bewegen Sie Paule in Blickrichtung nach vorne, bis er eine Wand erreicht.
- 6. Auf Paules Startfeld liegt bereits ein Korn. Lassen Sie Paule das Korn aufheben.
- 7. Bewegen Sie Paule über das Spielfeld und lassen Sie ihn dabei Körner aufheben.
- 8. Heben Sie 5 Körner auf dem Spielfeld auf. Lassen Sie Paule anschließend alle aufgesammelten 5 Körner auf ein Feld ablegen.
- 9. (Optional) Bewegen Sie Paule vor eine Wand. Lassen Sie ihn nun einen weiteren Schritt nach vorne laufen. Beschreiben Sie das Verhalten des Programms. Hinweis: Sie müssen nicht verstehen oder erklären können, was technisch passiert ist. Dies werden Sie noch im Laufe des Semesters lernen.

B.2 Objekte verstehen (0 Punkte)

In dieser Aufgabe sollen Sie Objekte im Hamster-Simulator, deren Eigenschaften, sowie deren Beziehung identifizieren.

- 1. Identifizieren Sie mindestens vier Objekte.
- 2. Überlegen Sie sich für jedes Objekt mind. 1 Attribut / Eigenschaft.
- 3. Überlegen Sie sich, wie die Objekte miteinander in Beziehung stehen.

Aufgabe C: Erste Java-Schritte mit Paule

(5 Punkte)

C.1 Let's Move (5 Punkte)

Im Folgenden sollen Sie eine Operation schreiben, die Paule automatisch durch das Territorium von Aufgabe B.1 laufen lässt, **alle** Körner aufsammeln lässt und diese in seiner Höhle (oben rechts – Spalte 7, Zeile 1, bei 0 beginnend – siehe Abb. C.1) wieder ablegen lässt.

Bearbeiten Sie dazu die Klasse FirstHamsterGame. Öffnen Sie dazu den Quellcode-Editor für die Klasse. Ergänzen Sie den Code in der Methode run.

Wenn Sie den Hamster-Simulator starten, führt dieser Ihren Code automatisch aus und lässt Paule entsprechend durch das Territorium laufen. Starten Sie den Hamster-Simulator ggf. mehrmals neu, wenn Sie Ihren Code angepasst haben und die Auswirkung sehen wollen.

Hinweis 1: Zeilen, die mit // beginnen, sind Kommentare. Sie können diese löschen.

Hinweis 2: Sie können Methoden auf Paule mit folgender Syntax aufrufen: paule.methodenname(); wobei methodenname eine von Paules verfügbaren Methoden aus der UI ist, zum Beispiel paule.move().

Hinweis für Fortgeschrittene: Implementieren Sie die Methode turnRight() in der Klasse FirstHamsterGame. Sie können diese Methode anschließend verwenden, um Paule nach rechts drehen zu lassen.

C.2 Der sprechende Hamster (Herausforderung für Fortgeschrittene, 0 Punkte)

- 1. Erweitern Sie Ihren Code aus Aufgabe C.1 so, dass Paule nun nach jedem Aufheben und Ablegen eines Korns die Anzahl der Körner auf dem Territorium ausgibt. *Hinweis:* Sie brauchen die Anzahl der Körner nicht zu speichern, sondern können sie als Text fest im Quelltext kodieren.
- 2. Passen Sie Ihren Code aus dem vorherigen Aufgabenteil so an, dass Sie diesmal keine fest kodierten Werte, sondern eine Variable verwenden, um Paule nach jedem Aufheben und Ablegen eines Korns die Anzahl der Körner auf dem Territorium ausgeben zu lassen.



Abbildung C.1: Zeigt die Höhle/Ziel

