

Elektronische Scheinprüfung - Softwareentwicklung I

Allgemeine Hinweise bzw. Tipps

Das Live Programming findet mit den eigenen Laptops am 16. Dezember von 11:45-13:15 in den Räumen R1.007 und R.1008 statt. In beiden Räumen wird jeweils ein Professor die Aufsicht übernehmen, sodass ich zwischen diesen beiden Räumen hin und her laufen kann, um Fragen zu beantworten. Die Raumaufteilung erfolgt vor dem Start, seien Sie daher bitte pünktlich. Elektronische Hilfsmittel sind neben dem eigenen Rechner keine erlaubt (keine Handys und Tablets). Denken Sie an ein Ladegerät für Laptop bzw. an ausreichend aufgeladene Batterien. Während der Prüfung werde ich die Ausweise von allen teilnehmenden Personen prüfen, halten Sie dazu bitte einen Lichtbildausweis und Ihren Studierendenausweis bereit.

Laden Sie Ihre Lösung erst am Ende der Prüfung in den [GitHub Classroom Kara Schein](#) auf den eigenen `main` Branch hoch. Wenn beim Erstellen des GitHub Classrooms ein Fehler auftritt, können Sie die Aufgabenstellung und das Ausgangsprojekt auch aus Moodle herunterladen. Wenn beim abschließendem Hochladen ein Fehler auftritt, warten Sie bitte, bis ich oder ein betreuender Professor an Ihren Platz kommt, um das Hochladen gemeinsam durchzuführen. Verlassen Sie nicht den Raum, bevor Ihre Lösung auch sicher in GitHub angekommen ist.

Es gibt insgesamt 3 Aufgaben, die möglicherweise in kleinere Teilaufgaben gesplittet sind, damit Sie leichter sehen, dass Sie bereits Teilschritte erfolgreich gelöst haben. Jede der drei Aufgaben wird mit 20 Punkten bewertet. Wenn Sie bei einer Aufgabe hängen, machen Sie mit der nächsten Aufgabe weiter und kommen Sie später zurück. Verteilen Sie die Arbeitszeit von 90 Minuten in etwa gleich auf die 3 Aufgaben.

Wie bereits angekündigt, ist auch ein guter Programmierstil wichtig, daher können Sie jeweils bis zu 2 Punkte pro Aufgabe verlieren (10%), wenn Sie sich nicht an den Codingstyle von mir halten. Für diese Prüfung sind nur die Resultate aus der CheckStyle Analyse relevant, die Sie auch in IntelliJ sehen können (1 Minuspunkt pro 5 Warnungen). Ein automatisches Prüfen der Lösungen über die GitHub Funktionalität (Autograding) steht Ihnen während der Prüfung nicht zur Verfügung.

Zum sicheren Bestehen der Prüfung reicht es, 40 von 60 Punkten zu erreichen. Sollten Sie die Prüfung nicht bestehen, oder an dem Prüfungstag krank sein (ärztliches Attest erforderlich), erhalten Sie am 13.1.2023 eine weitere Möglichkeit, den Schein zu erwerben.

Die Aufgaben müssen alleine und ohne fremde Hilfe gelöst werden. Sie dürfen zur Bearbeitung die Vorlesungsunterlagen zurate ziehen und auch Internetquellen aufrufen. Chats, Mails und andere Kommunikationsmittel sind nicht erlaubt. Beachten Sie, dass Sie keine Quelltexte aus fremden Quellen in Ihre Arbeit integrieren dürfen, das wäre sonst ein Plagiat. Alle Lösungen werden zudem nach der Prüfung in ein Plagiat-Prüfprogramm eingespielt: sollten dabei Duplikate entdeckt werden, werden die Arbeiten aller Beteiligten (also auch die Person, die das Original verfasst hat) mit einer 5 benotet. Eine Teilnahme an der Wiederholungsprüfung ist in diesem Fall nicht möglich.

Beachten Sie wieder, dass in allen Aufgaben beliebige Welten (gemäß der Aufgabenstellung) zulässig sind. D.h. eine Aufgabe ist nur teilweise gelöst, wenn Ihr Programm nur für einige der abgebildeten Varianten funktioniert. Die vorhandenen Tests prüfen einige der möglichen Szenarien, sind also ein guter Indikator, wie gut Ihre Lösung ist. Achtung: die Testergebnisse lassen sich aber nicht direkt in Punkte umrechnen. Manche Tests sind jetzt schon grün, auch wenn Sie noch gar nicht begonnen haben. Diese Tests müssen abschließend immer noch grün bleiben. Bei meiner Korrektur im Nachgang können außerdem zusätzliche Tests zum Einsatz kommen, um die korrekte Lösung zu bewerten. Für alle Tests gilt: entscheidend für den Erfolg eines Tests ist die korrekte Darstellung der erwarteten Welt. Die Position von Kara ist nicht relevant.

Die Voraussetzungen sind auch wie bisher für alle Aufgaben: Instanzvariablen, Container und Utility Methoden aus `java.util` sind **nicht erlaubt**. Nur lokale Variablen (und ggf. Methodenparameter) sind zur Speicherung von Werten und Arrays erlaubt. Zweidimensionale Arrays dürfen verwendet werden, bringen bei den Aufgaben allerdings keinen Mehrwert.