

Elektronische Prüfung: Allgemeine Infos & Vorbereitung

Organisatorisches / Vorbereitungen

- Wir werden den praktischen Teil der Prüfung am Computer mit Hilfe des Lernsticks erledigen. Der Lernstick ist eine vorkonfigurierbare Linux-Distribution (keine Sorge, die Handhabung ist sehr einfach). Sie ist so eingestellt, dass weder auf die lokalen Speichermedien noch auf das Internet zugegriffen werden kann. Neben den beiden IDEs (IntelliJ und Eclipse, jeweils in der aktuellsten Version) sind noch die Java-Tutorials und die JavaDoc verfügbar.
- Wir verwenden OpenJDK 17.
- Die USB-Sticks haben zwei Anschlüsse: Type-A USB 3.1 und Type-C USB 3.1
- Von euch im Vorfeld getätigte Einstellungen gehen verloren, da die Sticks wieder überschrieben werden. Bei den IDEs sind die Standard-Einstellungen gesetzt
- Speichern Sie häufig! Es wird alle 2 Minuten ein Backup gemacht (aber natürlich nur von den gespeicherten Dateien). Auf dieses Backup hat jedoch nur der Dozent Zugriff.
- Wer eine eigene Lernstick-Kopie haben möchte, muss einen leeren Memory-Stick mitbringen. Empfohlene Modelle sind auf der Lernstick-Webseite aufgelistet (https://www.digitale-nachhaltigkeit.unibe.ch/dienstleistungen/lernstick/hardware/index_ger.html). Wir verwenden das 16 GB Modell «SanDisk Ultra Dual Drive USB 3.1».
- Auf den Lernsticks befindet sich eine Beispiel-Aufgabe importiert in Eclipse, VSCode und IntelliJ. Es wird im Unterricht jeweils Zeit zur Verfügung gestellt, so dass alle ihren Lernstick testen können. Die Lernsticks sind nummeriert, so dass jede Person immer mit dem gleichen Stick testen kann und mit diesem dann auch die Prüfung absolviert. Eine Woche vor der Prüfung kann nicht mehr mit dem „eigenen“ Stick getestet werden, da dann die finale Prüfungsversion draufgespielt wird. Jedoch habe ich noch zusätzliche Sticks, mit welchen getestet werden kann.
- Wer den Lernstick nicht im Vorfeld am eigenen Computer getestet hat, geht das Risiko ein, dass es an der Prüfung dann nicht funktioniert und die Prüfung ohne Computer erledigt werden muss.

Prüfung

- Nach Ablauf der Prüfungszeit darf nicht mehr am Computer getippt werden. Im Anschluss werden die Lösungen auf das Prüfungsblatt abgeschrieben. (Ich bin mir bewusst, dass dies sehr mühsam ist. Ich möchte jedoch nicht das Risiko eingehen, dass aufgrund von technischen Problemen die komplette Prüfung verloren geht).
- Besteht trotz der sicheren Lernumgebung der Verdacht auf Betrug, werden die entsprechenden Verdachtsfälle noch zu einer mündlichen Nachprüfung / Gespräch aufgeboten.
- Ich werde keine Unit-Test in die Prüfungs-Umgebung integrieren, aber in allen Klassen eine main-Methode vorbereiten, damit ihr damit eure Implementation ausprobieren könnt.
- Für jede Aufgabe wird eine separate Klasse verwendet.
- Dank der IDE habt ihr natürlich gewisse Unterstützung wie Autovervollständigung. Auf der anderen Seite kann es natürlich sein, dass euch die IDE Kompilier-Fehler anzeigt oder ihr dank der main-Methode merkt, dass eure Implementation nicht ganz korrekt ist. Das ist die Kehrseite, denn auf Papier wird das nicht so schnell bemerkt. Da müsst ihr dann den Mut haben zur nächsten Aufgabe zu springen, wenn ihr das Gefühl habt, dass es grundsätzlich so funktionieren müsste, damit nicht zu viel Zeit verloren geht.
- Bei jeder Aufgabe müssen die zu bearbeitenden Dateien umbenannt werden! Hängen Sie bei den zu bearbeitenden Dateien ihren Namen anstelle von «Stud» (z.B. MyLinkedList_Stud.java => MyLinkedList_MichaelHenninger.java). Verwenden Sie keine Umlaute! Machen Sie sich dazu

mit der „File Renaming“ Funktion ihrer bevorzugten IDE vertraut (meistens Datei mit Rechtsklick anwählen und dann unter „Refactoring“ den entsprechenden Eintrag suchen).

- Verändern Sie keine bestehenden Methoden und keine Visibilitäts-Schlüsselwörter (private, public...), sofern nichts angegeben ist