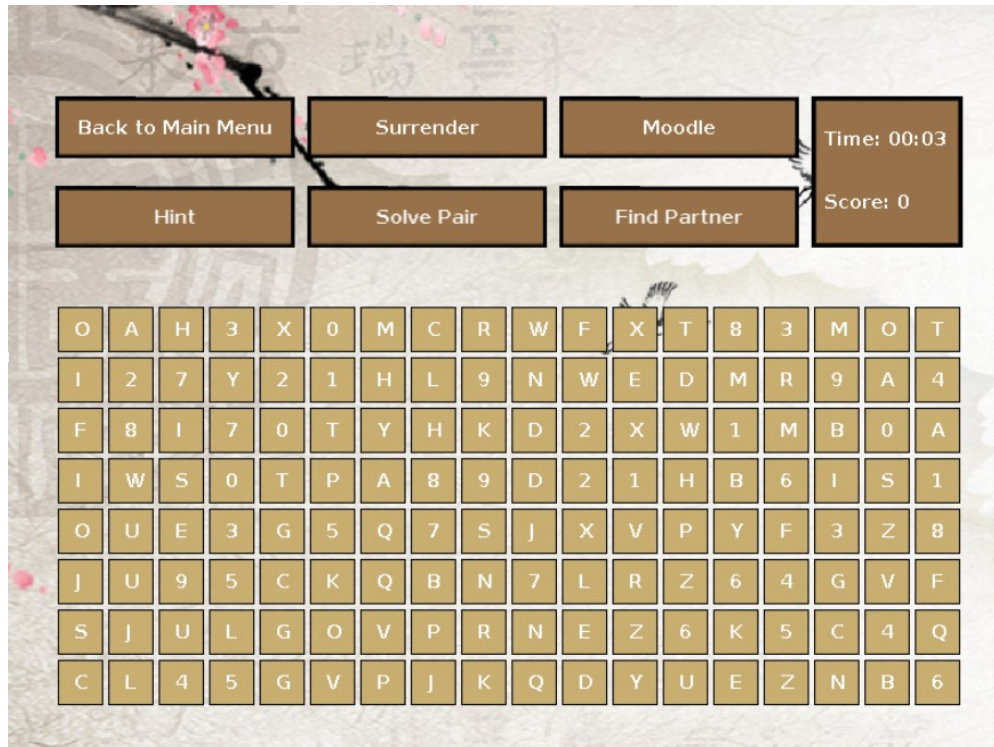


Allgemeine Informatik 1 (AI1)



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Sommersemester 2023 Programmierprojekt Organisatorisches und Inhalt



Dr.-Ing. Florian Stock

**Technische Universität Darmstadt
Fachbereich Informatik
FG Eingebettete Systeme und ihre Anwendungen**

**Hochschulstraße 10
D-64289 Darmstadt**

<https://www.esa.informatik.tu-darmstadt.de/>

Vorstellung

- Organisatorisches
- Aufgabenstellung
- Details
- Fragen

Allgemeine Informatik 1 Sommersemester 2023

Programmierprojekt, Version: 18. Mai 2023

Bearbeitungszeit: 07.06.2023 bis 05.07.2023



Organisatorisches

Wichtig: Lesen Sie sich zunächst die folgenden Formalitäten genau durch. Sollten diese nicht exakt beachtet werden, können wir Ihre Abgabe unter Umständen nicht werten. Lesen Sie die Aufgabenstellung vollständig durch, bevor Sie mit der Implementierung beginnen.

- Beachten Sie alle Hinweise zum Programmierprojekt auf der Seite:
<https://moodle.informatik.tu-darmstadt.de/mod/assign/view.php?id=55356>
- Nutzen Sie unsere **Vorgabe** und tragen Sie in der enthaltenen Datei **README.txt** Ihren Namen, Ihre Matrikelnummer und Ihre TU-ID (sowie bei Zweiergruppen die Daten Ihrer Partner*in!) ein.
- Bei Zweiergruppen reicht es aus, wenn die Lösung nur *einmal* hochgeladen wird. Sie sollten aber den*die Co-Autor*in in Moodle kennzeichnen.
- Abgabe des Projekts (spätestens bis zum **05.07.2023 um 22:00 Uhr**)
 - Verpacken Sie Ihr komplettes Projektverzeichnis in eine Standard-ZIP-Datei. Der Name muss aus den Initialen der Gruppenmitglieder und der Umgebung mit welcher Sie gearbeitet haben bestehen, also z. B. **SG_bluej.zip** bzw. **SG_eclipse.zip**. Bei Zweiergruppen entsprechend beide Namen, also z. B. **SG_DL_bluej.zip**.
 - Laden Sie die ZIP-Datei auf unserer moodle-Plattform hoch. **Keine** ZipX-, 7z- oder RAR-Archive! Eine genauere Anleitung zum Erstellen der Abgabe finden Sie auf Moodle im Bereich des Programmierprojekts.
 - Probieren Sie die Abgabe auf jeden Fall schon deutlich vor dem Abgabeschluss aus! Sie können mehrfach abgeben – bei mehrfachen Abgaben wird nur die zeitlich Letzte gewertet.

Ankündigungen und Termine

Wichtige Termine:

- Teilnahme am Programmierprojekt nur, wenn Sie für die Prüfung angemeldet sind!
- **07.06.2023:** Erste Erläuterung der Aufgabenstellung sowie Bereitstellung dieser auf der Lernplattform Moodle.
- **05.07.2023, 22:00 Uhr:** Abgabefrist Programmierprojekt!

Bewertung:

- Die Prüfung für diese Vorlesung besteht aus 2 Teilen:
 - Programmierprojekt (bis zu 25 Punkte)
 - Abschlussklausur (bis zu 75 Punkte)
- In Summe müssen Sie 50 Punkte erreichen, um zu bestehen!

!!! Wichtig: Das Programmierprojekt ist Teil der Prüfung !!!

- D.h. wir müssen und werden die Regeln strikt einhalten, um faire Bedingungen für alle Teilnehmer*innen zu garantieren.

Organisatorisches:

- Es soll möglichst in **Zweiergruppen** gearbeitet werden.
→ in Ausnahmen auch alleine (vorher bei Larissa abklären)
- Sie können im passenden Moodle-Forum nach einer*einem passenden Partner*in suchen.
- Studierende aus der Sammelgruppe wurden bereits aufgeteilt.
- Eine **TUCaN Anmeldung** ist erforderlich.
- Die **Abgabe** erfolgt über die Lernplattform Moodle.
- Nach dem Abgabeschluss werden die Lösungen bewertet.
- Die **Dokumentation des Quelltextes** fließt in die Bewertung ein.
- Abschließend finden **Testate** statt.

- Testate finden voraussichtlich vom **10.07.2022 - 28.07.2022 (Zoom)** statt.
- Die Übungsbetreuung fällt ab dem **10.07.** in diesem Zeitraum aus!

Ablauf:

- Der*Die Tutor*in stellt während des Testats Fragen zum Projekt.
- **Jedes Gruppenmitglied muss alle Aspekte der abgegebenen Lösung erläutern können!**
- Ansonsten gibt es für den*die jeweilige*n Studierende*n Punktabzüge bzw. im Extremfall gar keine Punkte.
- Genaueres zum Vorgehen werden wir auf Moodle ankündigen.

Wichtig: In Moodle stellen wir Updates & Infos bereit, falls das notwendig ist!

Arbeitsweise:

- Sie können eine IDE Ihrer Wahl benutzen. Wie empfehlen *BlueJ* oder *Eclipse*.
- Wichtig: Das Projekt **muss nach der Abgabe** direkt unter BlueJ oder Eclipse lauffähig sein und auch eine entsprechende **README-Datei** enthalten.

Hinweise zur Bewertung:

- Für Mängel bei Formatierung oder Kommentierung gibt es Abzüge.
- Nicht kompilierbare Abgaben werden mit **0 Punkten** bewertet.

Wichtig: In Moodle stellen wir Updates & Infos bereit, falls das notwendig ist!

Vorgehensweise (1/2):

- Laden Sie sich zunächst die Vorlage für dieses Programmierprojekt herunter. Sie enthält alle benötigten Klassen (minus den Klassen, die Sie implementieren werden).
- Wir werden detailliert instruieren, welche der Methoden der Klassen Sie sich intensiv ansehen sollten. Die meisten Klassen und Methoden sind schon vollständig.

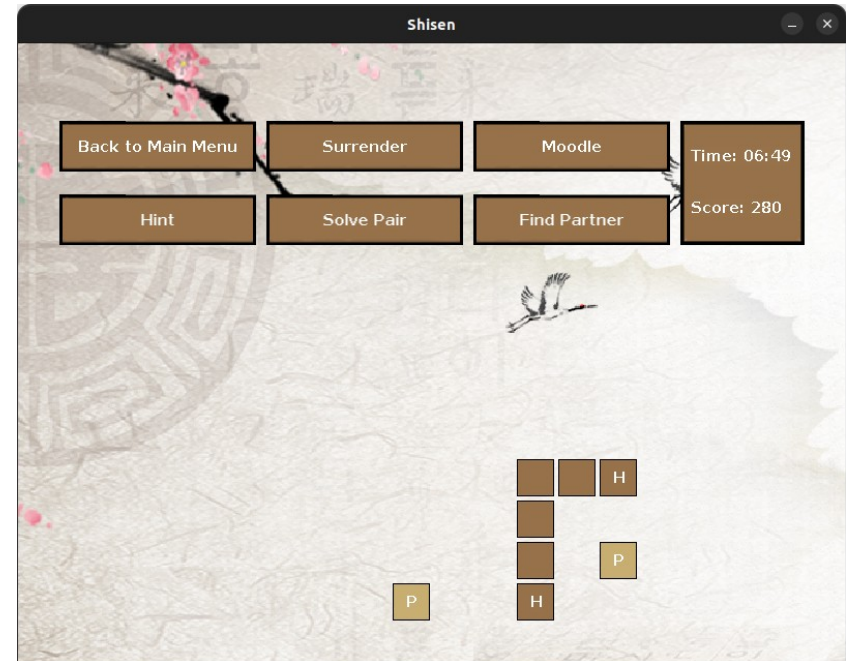
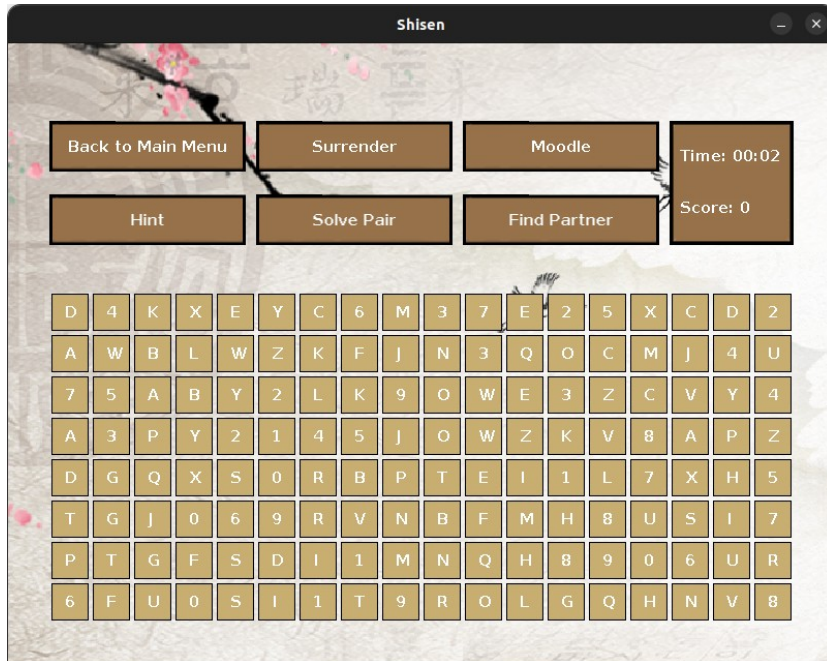
Wichtig: In Moodle stellen wir Updates & Infos bereit, falls das notwendig ist!

Vorgehensweise (2/2):

- Einige Methoden sind noch nicht implementiert bzw. enthalten nur eine „Dummy-Implementierung“, damit das Projekt kompiliert werden kann. Es ist im Folgenden Ihre Aufgabe diese Methoden zu vervollständigen.
- Alle anderen Methoden und Klassen der Vorgabe **müssen unangetastet bleiben**, auch sind **keine zusätzlichen Datenfelder** erlaubt (außer in eigenen Testklassen).
- Alle Methoden, die geändert werden müssen, enthalten TODOs und/oder sind in der Aufgabenstellung beschrieben.

Die Aufgabe: Shisen

- Es soll ein Brettspiel namens *Shisen* implementiert werden.
- Das Spiel ist eine 2D-Variante des Klassikers Mahjongg.
- Ziel des Spiels: Alle Feldpaare auf dem Spielfeld lösen.

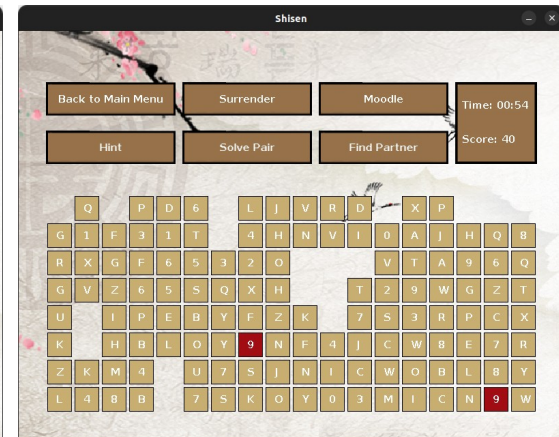
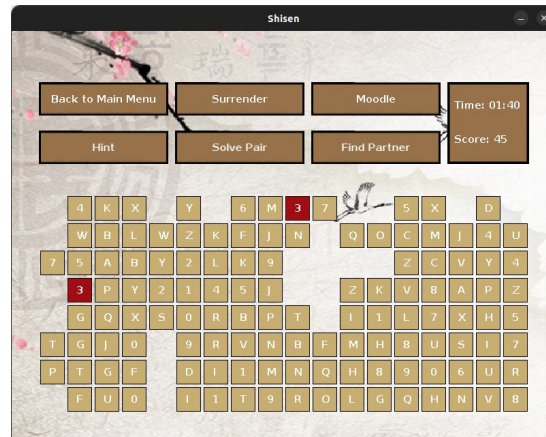
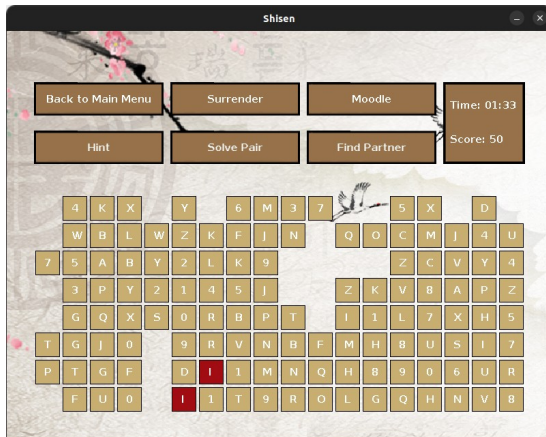
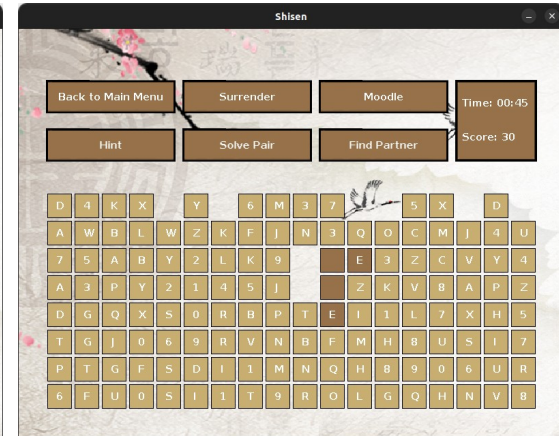
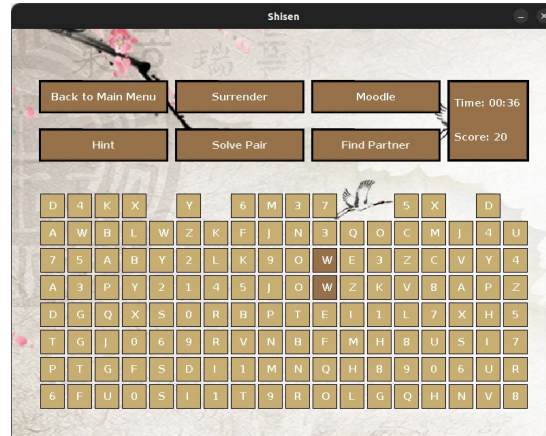
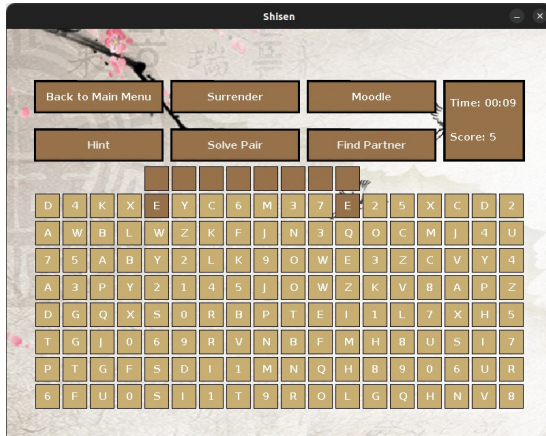


Spielregeln (1)

Im Spiel *Shisen* geht es darum alle Feld-Paare auf dem Spielfeld zu lösen und das in möglichst kurzer Zeit.

- Als Spielfeld dient eine Matrix, die eine feste Größe besitzt.
 - 8x18 (Felder), 10x20 (Felder mit Rand)
- Mit der Maus kann eine Karte ausgewählt werden:
 - Linksklick auf beliebiges Feld → Anwählen
 - Linksklick auf zweites Feld → Paar aktivieren
- Bedingung: Die Verbindung zwischen den beiden ausgewählten Feldern darf immer nur aus drei geraden Linien bestehen und keine anderen noch belegten Felder schneiden.

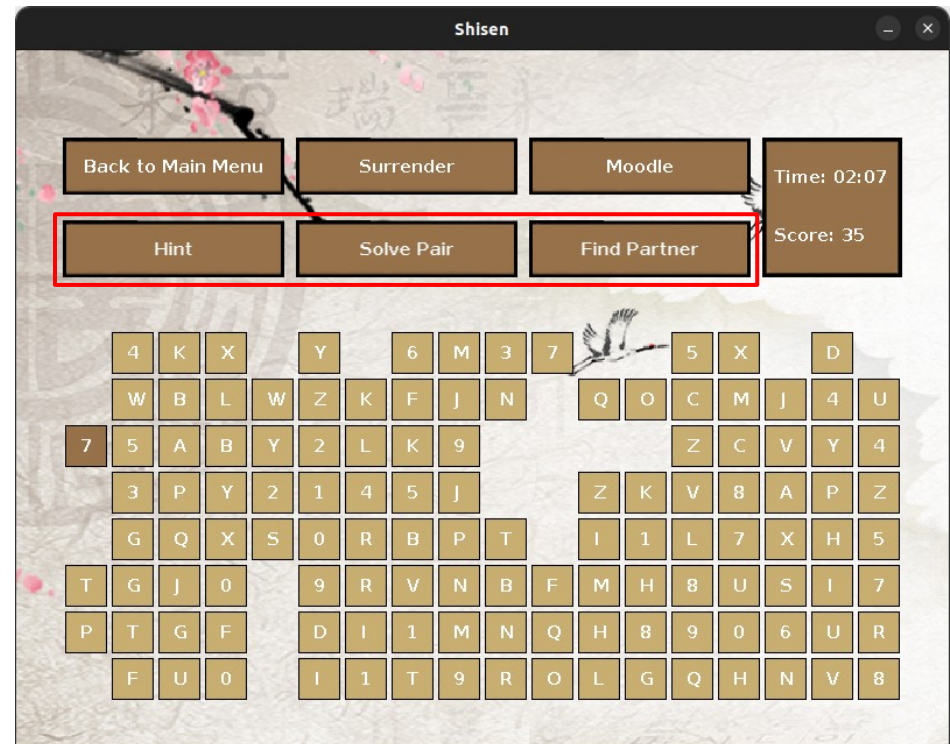
Spielregeln (2)



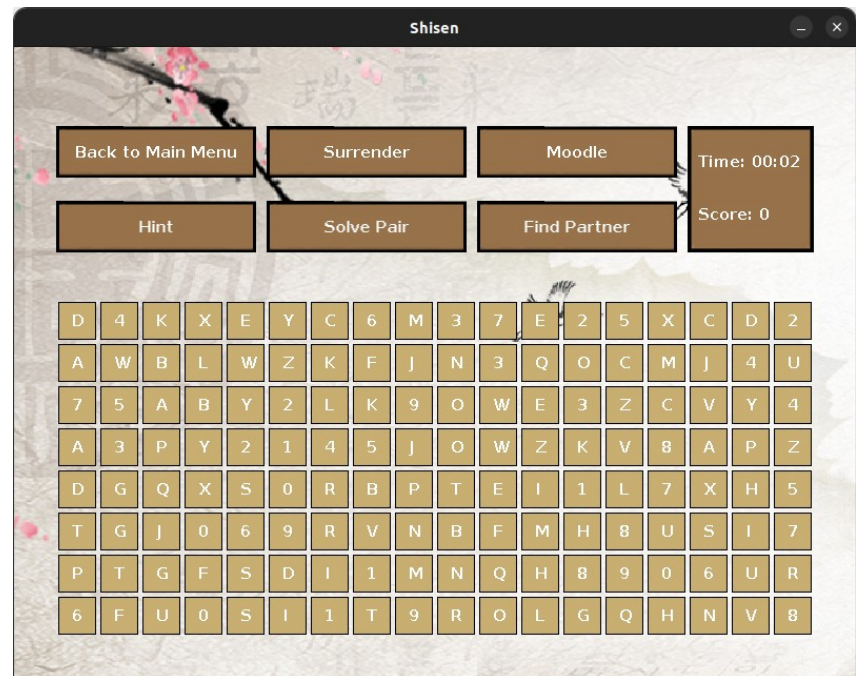
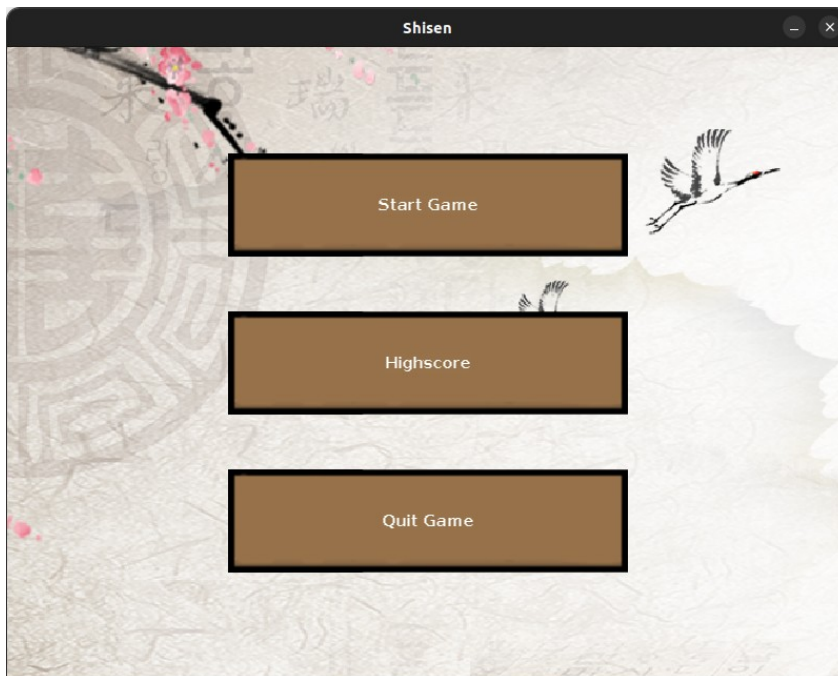
Zusatzfunktionen

Cheats:

- Es stehen bei erfolgreicher Implementierung drei Cheats zur Verfügung.
- Jeder Cheats zieht Punkte vom Score ab → Cheats können erst ausgeführt werden, wenn genug Punkte gesammelt wurden.



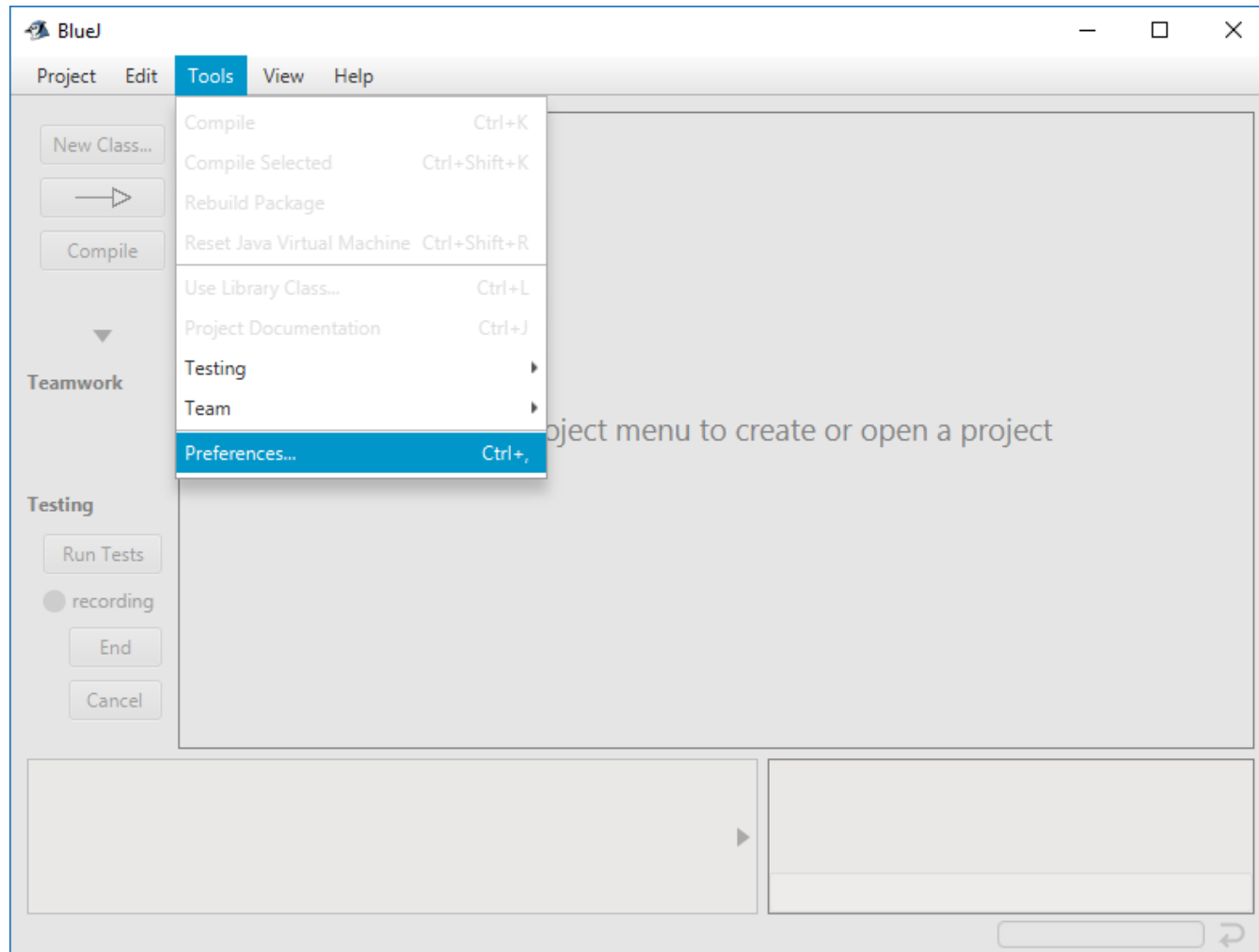
Live Demo



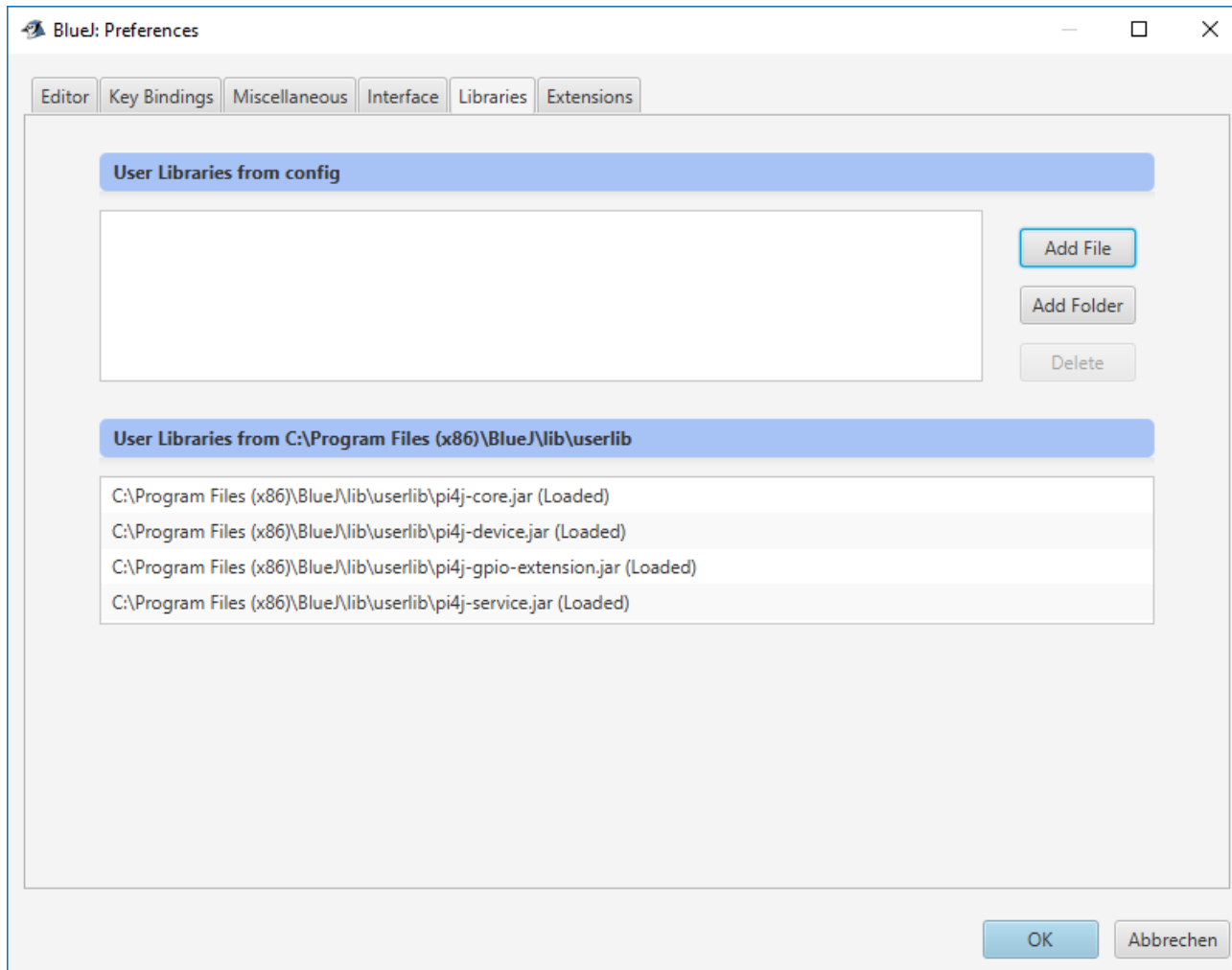
Projekt importieren in BlueJ

- Laden Sie das Gerüst herunter und entpacken Sie es anschließend:
<https://moodle.informatik.tu-darmstadt.de/mod/assign/view.php?id=55356>
- Öffnen Sie BlueJ und fügen Sie die benötigten Bibliotheken hinzu unter:
Tools → Preferences... → Libraries → Add File
- Gehen Sie dort zum entpackten Projektordner und fügen Sie die drei Bibliotheken hinzu: **eea.jar**, **lwjgl.jar** und **slick.jar**
- Anschließend müssen Sie BlueJ neustarten!
- Projektimport mit: *Project → Open Project...*

Projekt importieren in BlueJ



Projekt importieren in BlueJ



Projekt importieren in Eclipse

- Laden Sie das Gerüst herunter (nicht entpacken!):

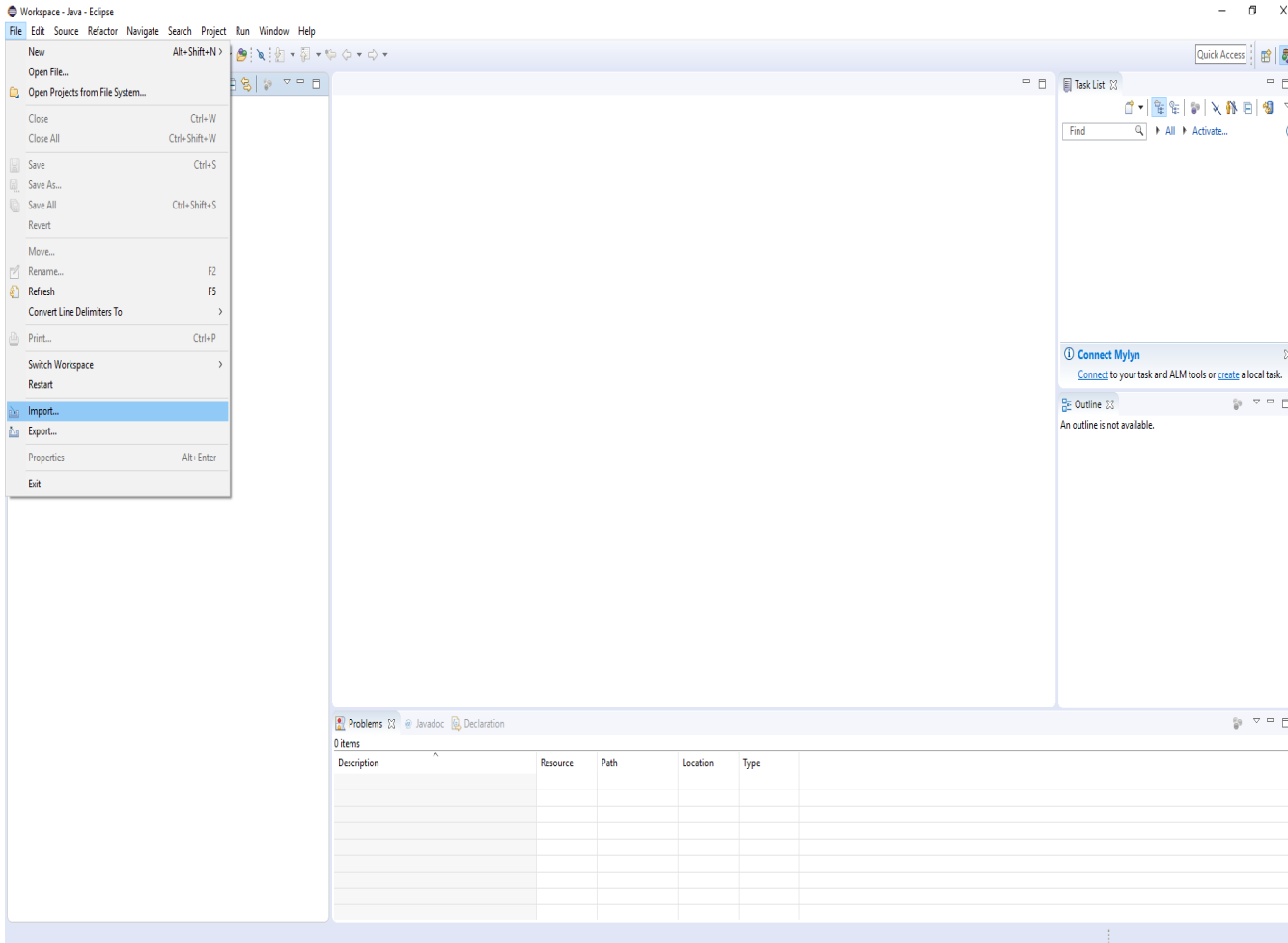
<https://moodle.informatik.tu-darmstadt.de/mod/assign/view.php?id=55356>

- Sie müssen das Projekt nur noch hinzufügen durch:

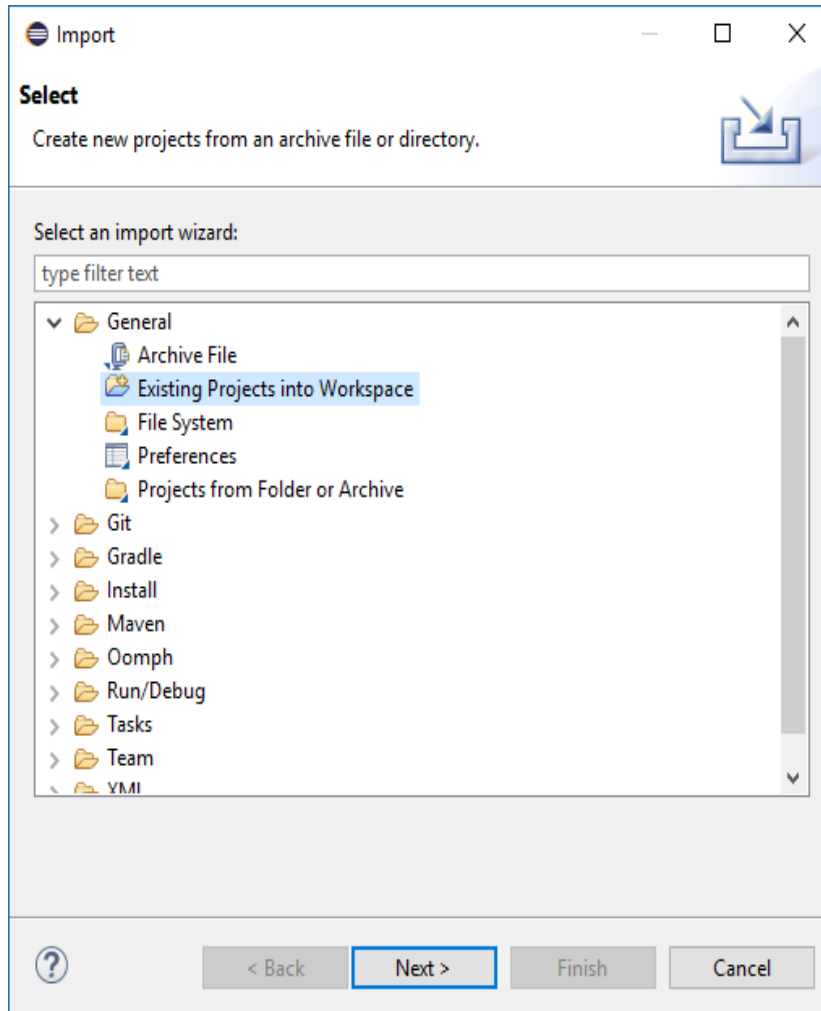
*File → Import → General → Existing Projects into Workspace
→ Select archive file → Browse*

- Fertig! Sie können direkt mit der Implementierung beginnen.
- Die „Welcome“-Seite können Sie direkt schließen.

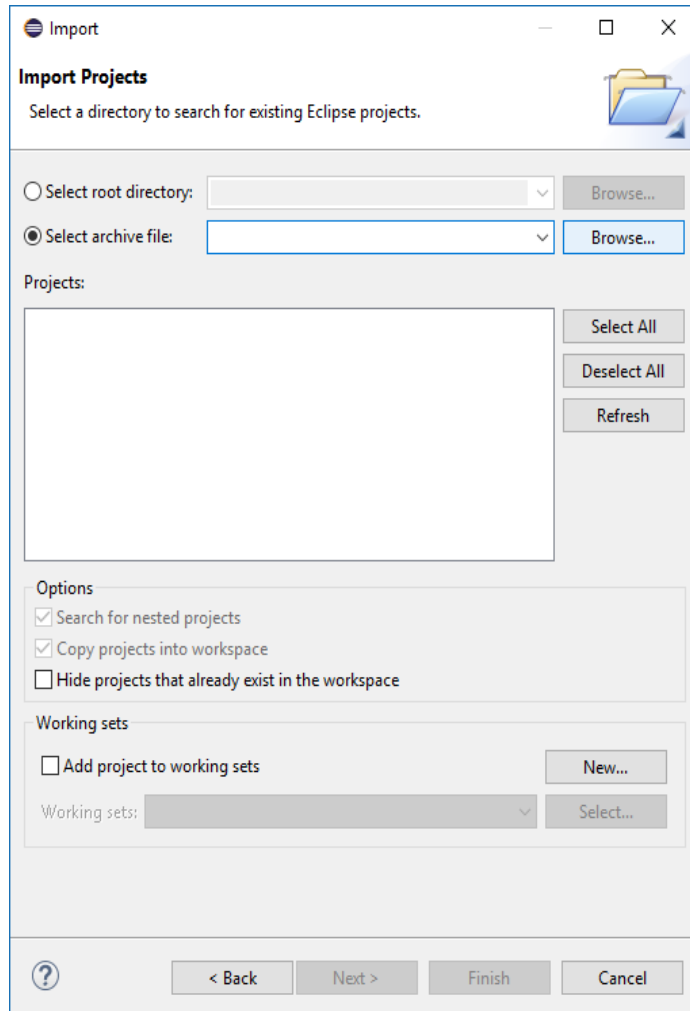
Projekt importieren in Eclipse



Projekt importieren in Eclipse



Projekt importieren in Eclipse



Projekt starten

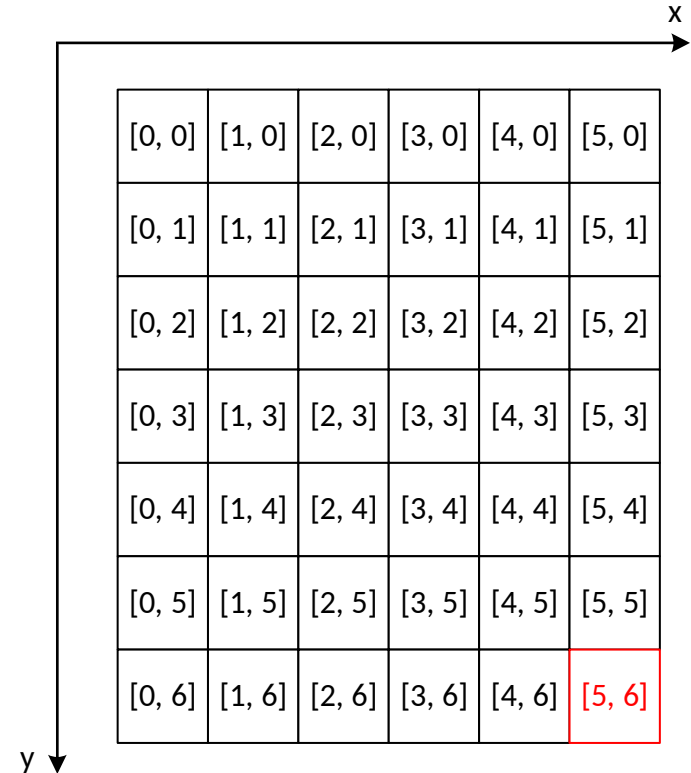
- Haben Sie die Aufgaben 1 und 2 vollständig implementiert, sollte sich das Programm kompilieren und starten lassen.
 - Achtung: Das Spiel kann zwar gestartet, aber noch nicht gespielt werden.
- Zum Starten müssen Sie die *main*-Methode der Klasse **Launch** ausführen.
 - Sie finden diese im Paket *view*.
 - Bei BlueJ sollte man das Paket *view* nicht öffnen bevor man Aufgabe 1 & 2 erledigt hat.

Anmerkungen zum Spielfeld

- In der Aufgabenstellung wird für die Darstellung einer Koordinate das Format **[x, y]** genutzt. Das Feld mit den Koordinaten $x=3$ und $y=1$ wird als `[3, 1]` dargestellt.
- Bei einem Zugriff auf ein Element im Spielfeld können Sie daher auch direkt `grid[x][y]` nutzen.

Anmerkungen zum Spielfeld

- Möchten Sie auf das rote Feld ganz unten recht zugreifen?
- Koordinaten sind $x=5$ und $y=6$
- Zugriff im Code erfolgt mit:
`grid[5][6]`



[0, 0]	[1, 0]	[2, 0]	[3, 0]	[4, 0]	[5, 0]
[0, 1]	[1, 1]	[2, 1]	[3, 1]	[4, 1]	[5, 1]
[0, 2]	[1, 2]	[2, 2]	[3, 2]	[4, 2]	[5, 2]
[0, 3]	[1, 3]	[2, 3]	[3, 3]	[4, 3]	[5, 3]
[0, 4]	[1, 4]	[2, 4]	[3, 4]	[4, 4]	[5, 4]
[0, 5]	[1, 5]	[2, 5]	[3, 5]	[4, 5]	[5, 5]
[0, 6]	[1, 6]	[2, 6]	[3, 6]	[4, 6]	[5, 6]

Anmerkungen zur Aufgabenstellung

- Sobald Sie mit der Implementierung der Highscore-Klassen (Aufgabe 4) beginnen, könnten Probleme mit der Highscore-Datei auftreten, da diese fehlerhaft beschrieben worden sein könnte.
- Sollten daher dort neue Fehler auftreten, dann könnten Sie zunächst versuchen die Datei “highscore.hs” zu löschen.
- Die Datei wird erneut erstellt, wenn ein neuer Highscore erreicht wurde.

- Nutzen Sie bei Fragen das *Forum zum Programmierprojekt* in Moodle
 - [Aufgabe] Frage
 - Bsp.: [5.1a] Wie ...
- **Keinen Code hochladen!**
- Wenn es Anliegen bezüglich des Programmierprojekts gibt, die nicht über das Forum in Moodle geklärt werden können, werden wir versuchen dies über Zoom in einer Art Sprechstunde zu klären.
- Bitte beachten Sie, dass das **Programmierprojekt Teil der Klausur** ist!
 - **Wir können Ihnen nur Hilfestellungen geben, aber keine Lösungen verraten.**

Programmierprojekt

Fällig: Mittwoch, 5. Juli 2023, 22:00



Hier finden Sie alle Informationen und Dateien rund um das Programmierprojekt.

Beachten Sie die Hinweise zur Abgabe in der *Aufgabenstellung.pdf* (aktuelle Version)!

Der Name des **ZIP**-Archivs soll wie folgt aufgebaut sein:

- Initialen_verwendeteProgrammierungsumgebung.zip
- **Beispiel:** LW_bluej.zip oder LW_eclipse.zip für eine Person und LW_RC_bluej.zip oder LW_RC_eclipse.zip bei einer Gruppe.

Die Abgabe muss direkt unter **BlueJ** oder **Eclipse** lauffähig sein!

	Aufgabenstellung.pdf	19. Mai 2023, 15:55
	Linux-Anleitung.pdf	19. Mai 2023, 15:55
	propro_bluej.zip	19. Mai 2023, 15:55
	propro_eclipse.zip	19. Mai 2023, 15:55

Mit der Abgabe ihrer Lösung bestätigen Sie, dass Sie/Ihre Gruppe der*die alleinige Autor*in/die alleinigen Autor*innen des gesamten Materials sind!

Plagiarismus ...

- wird als **Betrugsversuch** gewertet, d.h. die komplette Prüfung AI-1 gilt als nicht bestanden (5,0) auf Grund des Betrugsversuches.
- Wir werden nicht zwischen Urheber*in und Plagiator*in unterscheiden können, d.h. beide Arbeiten werden als Betrugsversuch gewertet.
 - Das bedeutet, dass „Abschreiben“ und „Abschreiben lassen“ gleichermaßen geahndet werden ... also keine gute Idee sind!
 - Lassen Sie also Ihren Rechner nicht unbeaufsichtigt und geben Sie z.B. keine USB-Sticks mit Ihren Projektdaten oder Quellcode weiter!

Mehr Infos zu Plagiarismus unter:

https://www.informatik.tu-darmstadt.de/studium_fb20/im_studium/studienbuero/plagiarismus/index.de.jsp

Bei weiteren Fragen

- Webseite: MOODLE Lernplattform
- <https://moodle.informatik.tu-darmstadt.de/course/view.php?id=1416>

Zur Übung

- Zuerst: Stellen Sie Fragen im Forum
- Kontaktieren Sie Ihren*Ihre Tutor*in ... (Zuteilung in MOODLE)
- Kontaktieren Sie die Haupttutorin:
 - Larissa Werner, larissa.werner@stud.tu-darmstadt.de

- **Zur Vorlesung**

- Zuerst: Stellen Sie Fragen im Forum
- Kontaktieren Sie die Veranstalter:
 - Leon Böck, boeck@tk.tu-darmstadt.de
 - Dr.-Ing. Florian Stock, stock@esa.tu-darmstadt.de