Stellen Sie bitte sicher, dass alles wie vorhergesehen läuft, bevor Sie dieses Übungsblatt abgeben. **Starten Sie den Kernel neu** (in der Menüleiste die Option Kernel→→Restart auswählen) und **validieren** Sie anschließend das Übungsblatt (in der Menüleiste auf Validate klicken) um Rückmeldung zu eventuellen fehlenden oder fehlerhaften Eingaben zu erhalten.

Füllen Sie alle Stellen im Übungsblatt aus, welche entweder DEIN CODE HIER oder "DEINE ANTWORT HIER" enthalten. Geben Sie unterhalb Ihren vollständigen Namen an.

Wenn Sie Code-Bestandteile aus anderen Quellen (wie z.B. Stackoverflow) kopieren, dann machen sie den kopierten Code in ihrer Quellcodedatei kenntlich und fügen eine Referenz auf die Quelle als Kommentar hinzu.

Wenn Sie die Aufgaben in einer Gruppe erledigen, dann fügen Sie die Namen aller Gruppenmitglieder in der nachfolgende Zelle zu Name und zusätzlich als Kommentar am Anfang Ihrer Quellcodedatei hinzu.

NAME **=** ""

|  |  |
| --- | --- |
|  | **Betriebssysteme**  **Wintersemester 2023/24** |

**Praktische Klausur**

Klausur am **Samstag, 14. Oktober 2023**

**Achtung:**

*Bitte speichern Sie Ihre Quellcode-Dateien (\*.c) im selben Ordner wie dieses Übungsblatt. Beachten Sie außerdem die Kommentare zur richtigen Benennung Ihrer Dateien. Nur so ist eine korrekte Abgabe und Auswertung Ihrer Lösungen möglich.*

**Händler und Jäger**

**Allgemeine Hinweise**

* Es sind alle Unterlagen (Foliensätze, Skripten) erlaubt, Sie dürfen auch Teile Ihrer eigenen Lösungen der Übungsblätter wiederverwenden.
* Kommunikation mit anderen Studenten und KIs ist in jeglicher Form verboten und führt zu einer negativen Beurteilung.
* Lesen Sie die Aufgabe genau und vollständig durch, bevor Sie mit der Bearbeitung beginnen.
* Zur Benotung wird die Funktionalität ihrer Lösung auf JupyterHub herangezogen.
* Kommentieren Sie Ihren Code ausführlich (jede Funktion und auch jeden wichtigen Codeteil)
* Kommentieren Sie am Anfang Ihrer Quelldatei, was an Ihrem abgegebenen Programm einwandfrei funktioniert, und was nicht funktioniert (und warum).
* Wann immer von „zufälligen“ Werten gesprochen wird, grenzen Sie den möglichen Wertebereich sinnvoll ein.
* Wenn keine konkreten Werte angegeben werden, dann wählen Sie selber einen sinnvollen Wert aus.

**Aufgabe**

Es war einmal ein reicher Händler aus Innsbruck, der sich in den Tiroler Bergen verirrte. Nachdem er mehrere Tage lang vergeblich umhergeirrt war, stieß der Händler schließlich auf eine Gruppe armer Jäger, die versuchten, über die Runden zu kommen. Nachdem die Jäger den Händler mit Lebensmitteln und Informationen versorgt hatten, damit er sicher nach Hause zurückkehren konnte, beschloss der Händler, die Jäger mit Goldmünzen zu belohnen. Daher beschloss der Händler, bevor er seines Weges ginge, alle Münzen aus seiner magischen Brieftasche an die Jäger zu verteilen. Die magische Brieftasche hat die folgenden Eigenschaften:

* sie kann nur eine begrenzte Anzahl an Münzen enthalten
* sobald eine Münze aus der Brieftasche entnommen wird, wird sie nach einer bestimmten Zeit durch eine neue Goldmünze ersetzt.

Der Händler wirft für 5 Minuten Münzen auf den Boden. Die Jäger kämpfen darum, die Münzen einzusammeln. Jede Münze kann nur von einem Jäger aufgenommen werden.

Ihre Aufgabe ist es, das Werfen und Einsammeln der Münzen zu simulieren und am Ende auszugeben, wie viele Münzen von jedem Jäger eingesammelt wurden.

Um Ihr Programm sauber schließen zu können, soll ein gracefull Shutdown implementiert werden, der die Threads sauber beendet.

**Abgabe**

Eine C-Quelldatei goldcoins.c und ein Makefile namens Makefile

Am Anfang Ihrer C Datei muss ein Kommentar stehen, welcher folgende Daten enthält:

* Ihren vollständigen Namen,
* Ihre Matrikelnummer

%%bash

*#*

*# Schreiben Sie ein Makefile, dass Ihren Quellcode kompiliert.*

*#*

*# Kopieren Sie vor der Abgabe ihr Makefile und Ihre Quellcodedatein in dasselbe Verzeichnis,*

*# in dem dieser Angabezettel zu finden ist.*

*#*

*# Der Name der ausführbaren Datei soll 'goldcoins' lauten (auf korrekte Groß- und Kleinschreibung achten!).*

*#*

​

*# Das Ausführen dieser Zelle sollte das Makefile ausführen*

make

gcc -c -Werror -Wall -lpthread -o goldcoins.o goldcoins.c

gcc -pthread -o goldcoins goldcoins.o