

Übungsblatt

**Technische Informatik 2**

Sommersemester 2025

Woche 1

**Unbedingt zu beachtende Hinweise:**

Ihre Lösungen werden zum Teil rechnerunterstützt ausgewertet. Hierzu ist es unbedingt erforderlich, dass Sie Ihre Lösung (ggf. inkl. Kommentierung) in elektronischer Form an entsprechender Stelle in diesem PDF eingeben. Verwenden Sie hierzu einen PDF-Reader, mit dem Sie Ihre Lösung als Text (und nicht in einer anderen Form, also auch nicht als Anmerkung/Annotation) in die hierfür vorgesehenen Felder schreiben können. Derartige PDF-Reader unterstützen das Ausfüllen sogenannter PDF-Formulare (engl. forms).

Aufgaben

1. In der Vorlesung wurden u.a. fünf Eigenschaften eingebetteter Systeme vorgestellt. Geben Sie nachfolgend diese Eigenschaften stichwortartig an.

10 P
------

- (a) (2 P) Eigenschaft Nr. 1:

Ein oder mehrere digitalen Rechen- respektive Verarbeitungseinheiten

- (b) (2 P) Eigenschaft Nr. 2:

Sie haben ein Ein-/Ausgang

- (c) (2 P) Eigenschaft Nr. 3:

Bestandteil eines umgebenden technischen Systems

- (d) (2 P) Eigenschaft Nr. 4:

Interagieren über Sensoren/Aktoren

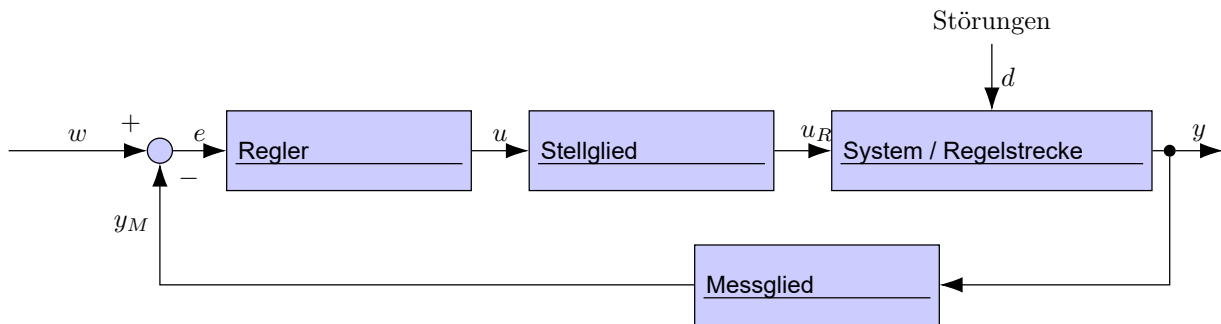
- (e) (2 P) Eigenschaft Nr. 5:

Verfolgen eine sehr konkrete Aufgabe

2. Nachfolgend abgebildet ist eine Skizze eines Regelkreises.

(a) (4 P) Beschriften Sie die blau hinterlegten Blöcke

6 P



(b) (2 P) Erläutern Sie nachfolgend die Bedeutung der in obiger Skizze zu findenden Größe  $y_M$  in stichwortartiger Form.

$y_M$  ist der Istwert(aktuelle gemessene Wert), welcher ein wichtiger Bestandteil ist, weil man durch die Überwachung abweichungen vom Sollwert erkennt.

Es wird die Regeldifferenz dann berechnet, welche an den Regler weitergegeben.

3. Nachfolgende Aufgaben beziehen sich auf die Ihnen in der ersten Übung vermittelten Grundlagen der Programmierung in C.

5 P

(a) (2 P) Beschreiben sie die semantische Bedeutung der Parameter, die `main()` bei einem Programstart übergeben werden.

`main()` ist der Einstieg in das Programm.

`int argc`: Anzahl der Elemente in `argv`.

`char* argv[]` ist ein Array von Argumenten, welche dem Programm übergeben werden. Außerdem ist `argv` der eigene Aufrufname

(b) (1 P) Ergänzen Sie nachfolgenden Lückentext in fachlich korrekter Weise:

Bei der Beendigung eines (C-)Programms ohne Fehler sollte dieses mit dem Returncode 0 terminieren.

(c) (2 P) Geben eine (!) C-Programmzeile an, die – nach Übersetzung und Ausführung – dazu führt, die Größe des zu Ablage eines `unsigned int` benötigten Speichers auf dem ausführenden System auszugeben.

`printf("%d", sizeof(unsigned int));`

4. Ergänzen Sie den in den Unteraufgaben jeweils weitergeführten Text in fachlich korrekter Weise:

12 P
------

- (a) (1 P) Sie befinden sich in einem Git-Repository und sollen sich die Arbeit eines Kollegen auf dem Branch `test` anschauen.
- (b) (1 P) Mit dem Befehl `git branch` \_\_\_\_\_ können Sie alle lokalen Branches anzeigen lassen.
- (c) (1 P) Der aktuelle Branch wird dabei mit einem `*` \_\_\_\_\_ markiert.
- (d) (1 P) Stellen Sie sicher, dass Sie sich auf dem Branch `main` befinden (ggf. mit `git checkout main` \_\_\_\_\_ dorthin wechseln).
- (e) (1 P) Anschließend verwenden Sie `git diff main test` \_\_\_\_\_, um die Unterschiede zwischen Ihrem aktuellen Branch und `test` anzuzeigen.
- (f) (1 P) Der `main`-Branch ist in vielen Projekten geschützt, sodass keine direkten Änderungen erlaubt sind. Zudem dient er oft als stabiler Entwicklungsstand. Durch separate Branches (z.B. `test`) können Sie unabhängig daran arbeiten, ohne die Hauptentwicklung zu gefährden. Die Ergebnisse können später mit `git merge <test>` \_\_\_\_\_ wieder zusammengeführt werden.
- (g) (1 P) In diesem Repository existiert eine Datei namens `README.md`. Mit `git log` \_\_\_\_\_ können Sie sehen, wann und von wem Änderungen an der Datei vorgenommen wurden.
- (h) (1 P) Wenn Sie zusätzlich die konkreten Änderungen einsehen möchten, verwenden Sie `git diff README.md` \_\_\_\_\_. (Anmerkung: der gesuchte Befehl wird einmal ohne und einmal mit Optionen abgefragt.)
- (i) (1 P) Nachdem Sie einen Rechtschreibfehler in `README.md` korrigiert haben, möchten Sie diese Änderung übertragen: Zuerst prüfen Sie mit `git status` \_\_\_\_\_, ob Änderungen erkannt wurden.
- (j) (1 P) Dann fügen Sie die korrigierte Datei mit `git add README.md` \_\_\_\_\_ zur Staging-Area hinzu.
- (k) (1 P) Anschließend erstellen Sie mit `git commit -m "Nachricht"` \_\_\_\_\_ einen Commit.
- (l) (1 P) Zum Schluss übertragen Sie die Änderung mit `git push` \_\_\_\_\_ ins Remote-Repository.