

Programmierfallen – unzulässige Kopplung

#1 unzulässige Kopplung von Zustand mit Programmablauf

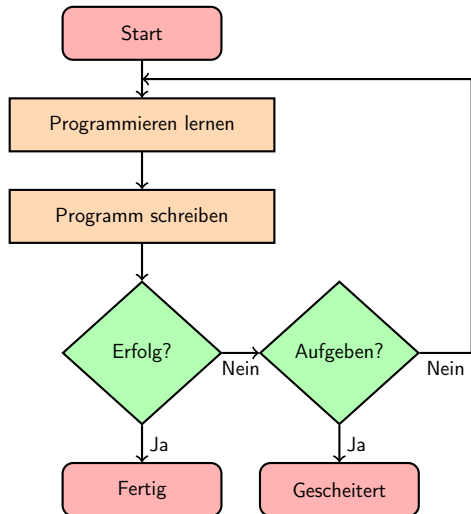
Stefan Helmert

entroserv.de

09/2019

entroserv
course

Ausgangssituation



- ▶ Erlernen von Programmiersprachen
- ▶ Anschauen von Tutorials, Beispielen
- ▶ Beispielprogramme leicht umsetzbar
- ▶ reale Lösung nicht umsetzbar

Ausgangssituation

Anforderungsanalyse

Problem

Vermeintlicher Konflikt

Problemanalyse

Nur Anforderung A

Beide Anforderung A und B

Lösungsansatz

Scheduling

Zustandsautomat

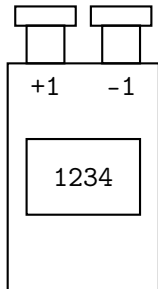
Lösung

Zustandsvariablen

Kontakt

Ausgangssituation

Anforderungsanalyse

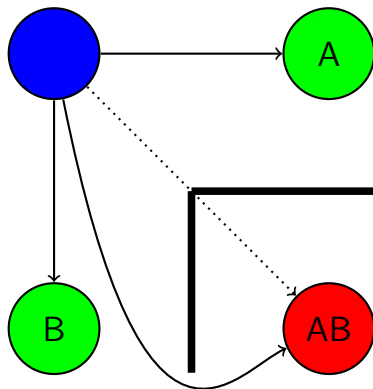


Aufgabe

- ▶ zähle alle Tastendrucke
- ▶ Taste A erhöht um 1
- ▶ Taste B reduziert um 1
- ▶ nur ein Thread
- ▶ keine asynchrone Programmierung

Problem

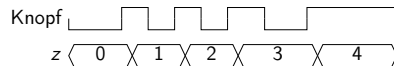
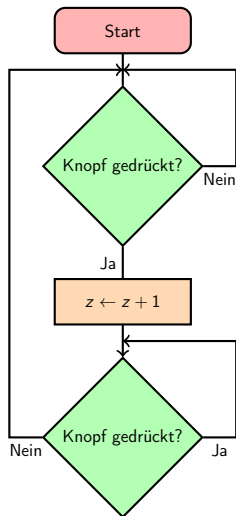
Vermeintlicher Konflikt



- ▶ Anforderung A
 - ▶ unmittelbar umsetzbar
- ▶ Anforderung B
 - ▶ unmittelbar umsetzbar
- ▶ Kombination aus A, B
 - ▶ komplexe Lösung

Problemanalyse

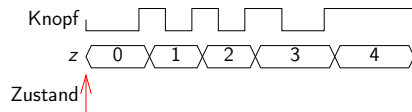
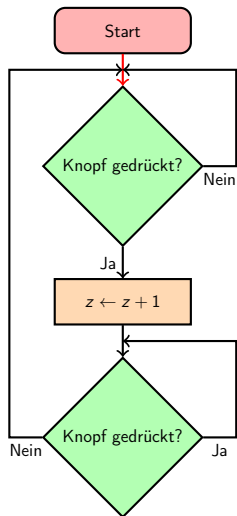
Nur Anforderung A



► zählt Tastendrucke: z

Problemanalyse

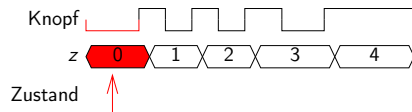
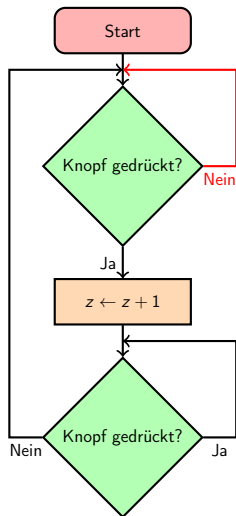
Nur Anforderung A



- Knopf ist nicht gedrückt
- Programmstart

Problemanalyse

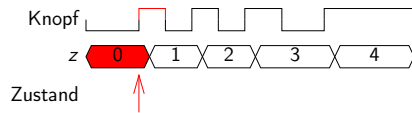
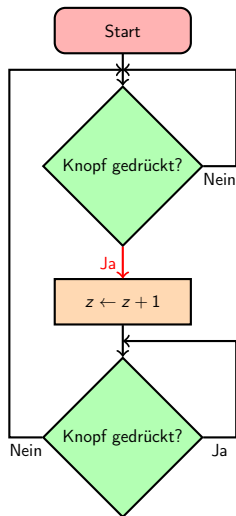
Nur Anforderung A



- Knopf ist nicht gedrückt
- Warteschleife solange Knopf nicht gedrückt ist

Problemanalyse

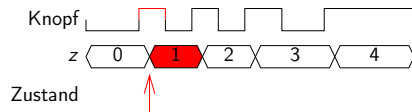
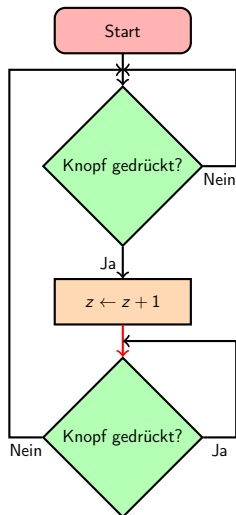
Nur Anforderung A



- Knopf ist gedrückt
- Warteschleife wird verlassen

Problemanalyse

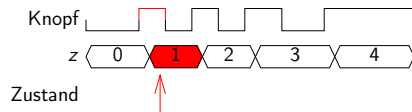
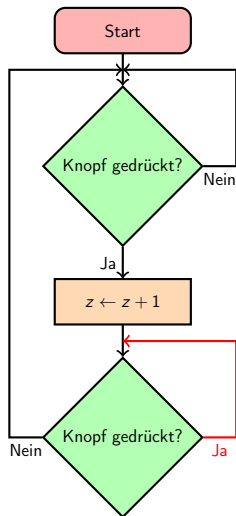
Nur Anforderung A



- Zählvariable z erhöht um 1
- Beginn der Wartschleife bis Knopf losgelassen

Problemanalyse

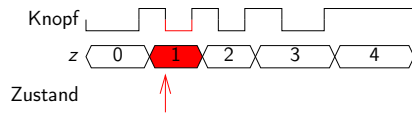
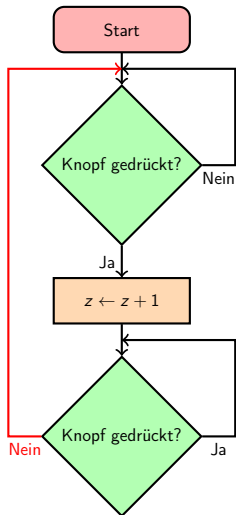
Nur Anforderung A



► Wartschleife bis Knopf losgelassen

Problemanalyse

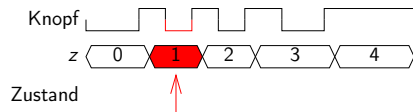
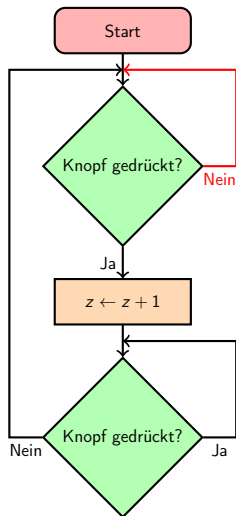
Nur Anforderung A



- ▶ Knopf losgelassen
- ▶ Wartschleife wird verlassen
- ▶ erneuter Aufruf der ersten Wartschleife

Problemanalyse

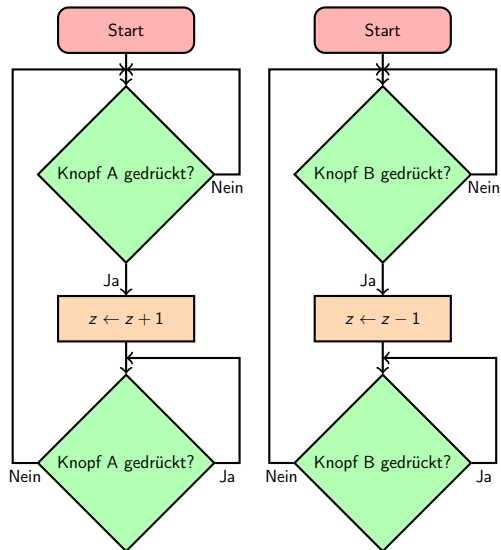
Nur Anforderung A



- Knopf nicht gedrückt
- Wartschleife bis Knopf gedrückt

Problemanalyse

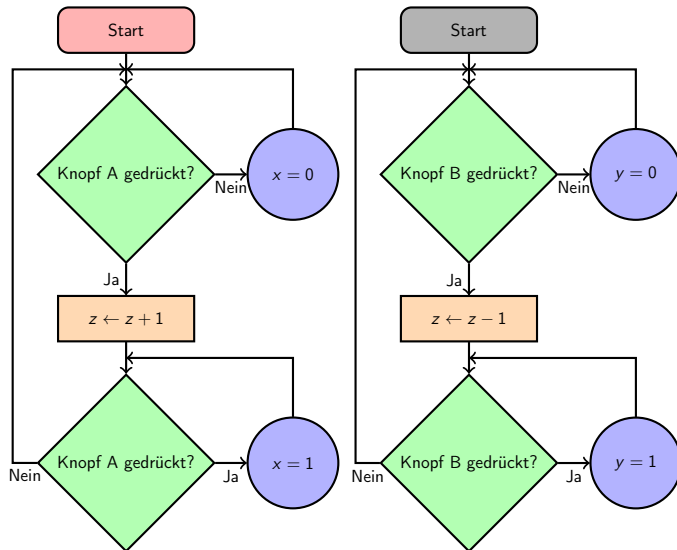
Beide Anforderung A und B



- ▶ Flussdiagramm duplizieren?
- ▶ aber Threads nicht erlaubt

Lösungsansatz

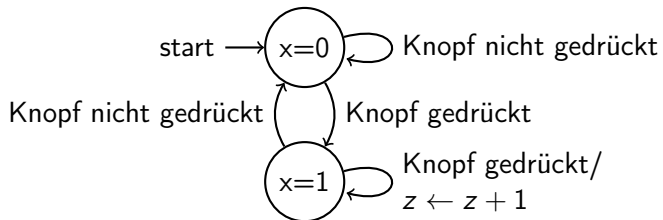
Scheduling



- ▶ implementiere Multithreading
- ▶ definiere Schedulingpunkt
- ▶ speichere Rücksprungpunkt
- ▶ Nachteile
 - ▶ komplex
 - ▶ fehleranfällig

Lösungsansatz

Zustandsautomat



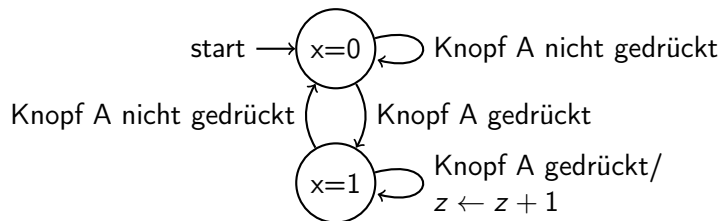
Zustandsautomat

- ▶ Betrachtung der Zustände
- ▶ Befehlsabfolge zunächst irrelevant

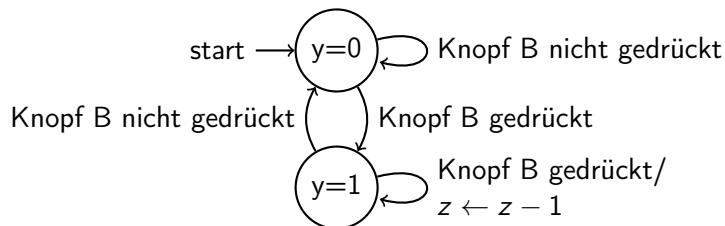
Lösungsansatz

Zustandsautomat

Zustandsautomat für Knopf A

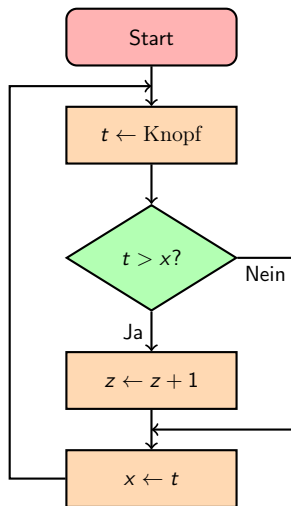


Zustandsautomat für Knopf B



Lösung

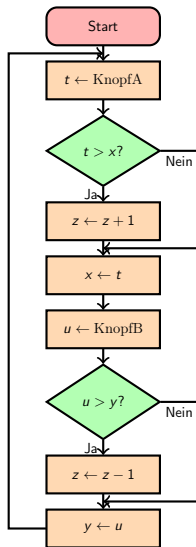
Zustandsvariablen



- ▶ Zustand von Programmablauf getrennt
- ▶ Zustandsvariable x
- ▶ nur eine Hauptschleife

Lösung

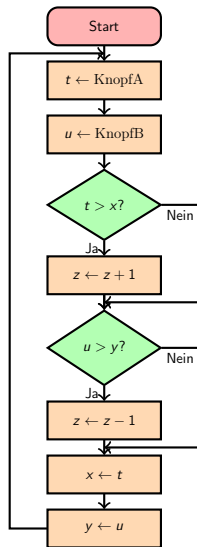
Zustandsvariablen



- ▶ erfolgreiche Kombination beider Anforderungen
- ▶ Zustand von Programmablauf getrennt
- ▶ Zustandsvariablen
 - ▶ x für Knopf A
 - ▶ y für Knopf B
- ▶ nur eine Hauptschleife
 - ▶ nur ein Sprung zurück

Lösung

Zustandsvariablen



- Umsortieren nach EVA-Prinzip
 - Wartbarkeit

Kontakt

Folien <https://github.com/TheTesla/ProgrammierTutorial>

Github <https://github.com/TheTesla>

Webseite <https://entroserv.de/>

Mailingliste <https://www.lists.entroserv.de/listinfo/lounge>

E-Mail stefan@entroserv.de

Telegram [@Tesla423](#)