

Übungen zu Systemprogrammierung 1

Gitlab CI

Sommersemester 2025

Luis Gerhorst, Thomas Preisner, Tobias Häberlein, Jürgen Kleinöder

Lehrstuhl für Informatik 4
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Informatik 4
Systemsoftware



Friedrich-Alexander-Universität
Technische Fakultät

A **aufgabe5** 
Project ID: 25215 

  Star 0  Fork 0

 1 Commit  1 Branch  3 Tags  21 KB Project Storage

Topics: aufgabe5 rabenstein

 **init from 4fe7dd4e**  a65f6391 
Systemprogrammierung authored 2 weeks ago

main aufgabe5 / +

Find file Web IDE  Clone

- **Continuous Integration**
- Pipeline mit vordefinierten Checks/Tests
- Ausführung nach einem Push von neuen Commits
 - ⇒ Automatisiertes Testen
 - ⇒ Entdecken von Fehlern
 - ⇒ Überblick über Stand der Softwarefunktionalität



- Pipeline, bestehend aus Jobs und Stages
- Stages...
 - gruppieren mehrere Jobs
 - definieren, wann welche Jobs ausgeführt werden
 - z.B. build oder test Stage
- Jobs ...
 - definieren, was zu tun ist
 - werden von Runnern ausgeführt
- Tests ...
 - als Teilergebnisse von Jobs
 - liefern Rückmeldung, wie erfolgreich die Tests für den gegebenen Commit waren

Hands-On: Übersicht über mehrere Pipelines



i4sp > ... > rabenstein > aufgabe5 > Pipelines

All **1** Finished Branches Tags

Clear runner caches

CI lint

Run pipeline

Filter pipelines



Show Pipeline ID ▾

Status

Pipeline

Triggerer

Stages

failed

init from 4fe7dd4e

#105574 main a65f6391



🕒 00:00:11

📅 55 minutes ago

latest



✖ failed Pipeline #105574 triggered 57 minutes ago by Eva Dengler

Retry

Delete

init from 4fe7dd4e

🕒 2 jobs for `main`
in 11 seconds and was queued for 2 seconds

🚩 `latest`

🔗 `a65f6391`

🔗 No related merge requests found.

Pipeline Needs Jobs **2** Failed Jobs **1** Tests **8**

build

test



Hands-On: Jobs einer Pipeline



failed Pipeline #105574 triggered 57 minutes ago by Eva Dengler

Retry

Delete

init from 4fe7dd4e

🕒 2 jobs for `main`
in 11 seconds and was queued for 2 seconds

🚩 latest

🔗 a65f6391

🔍 No related merge requests found.

Pipeline Needs **Jobs 2** Failed Jobs 1 Tests 8

Status	Job	Stage	Name	Duration	Coverage
failed	#897100 🚩 main 🔗 a65f6391	test	test	🕒 00:00:06 🕒 57 minutes ago	
passed	#897099 🚩 main 🔗 a65f6391	build	build	🕒 00:00:04 🕒 57 minutes ago	

Hands-On: Tests einer Pipeline



failed Pipeline #105574 triggered 57 minutes ago by Eva Dengler

Retry

Delete

init from 4fe7dd4e

🕒 2 jobs for `main`
in 11 seconds and was queued for 2 seconds

🚩 latest

🔗 [a65f6391](#)

🔗 No related merge requests found.

Pipeline Needs Jobs 2 Failed Jobs 1 **Tests 8**

Summary

8 tests 7 failures 0 errors 12.5% success rate 0.00ms

Jobs

Job	Duration	Failed	Errors	Skipped	Passed	Total
test	0.00ms	7	0	0	1	8

Hands-On: Detailansicht der Tests einer Pipeline



< test

8 tests 7 failures 0 errors 12.5% success rate 0.00ms

Tests

Suite	Name	Filename	Status	Duration	Details
	calloc_calls_malloc		✘	0.00ms	View details
	calloc_js_zeroed		✘	0.00ms	View details
	calloc_too_much_fails		✘	0.00ms	View details
	free_valid_pointer_should_succeed		✘	0.00ms	View details
	malloc_max_size_succeeds		✘	0.00ms	View details
	realloc_calls_malloc		✘	0.00ms	View details
	realloc_too_much_fails		✘	0.00ms	View details
	malloc_too_much_fails		✔	0.00ms	View details



Scenario: student implementation creates the same output as the reference implementation

Given I have a directory “/tmp/creeper“ with the following structure

... passed in 0.019s

name	type	
dir1/subdir1/	directory	
dir1/subdir1/super_source_file.c	file	
dir1/subdir1/secret_source_file.c	file	
dir1/subdir1/Makefile	file	
dir1/subdir2/	directory	
dir2/subdir2/	directory	

And I add “/tmp/creeper“ to the list of paths ... passed in 0.000s

When I run creeper ... passed in 0.003s

And I run creeper's reference implementation ... passed in 0.003s

Then student and reference implementations create the same output

... failed in 0.000s

- **Merke:** failed zeigt, welcher Schritt fehlgeschlagen ist
- **Keywords aus Behavior Driven Development (BDD)**
 - ⇒ Given: Vorbedingung
 - ⇒ When: Ausführung
 - ⇒ Then: Erwartetes Ergebnis



Testcases:

- Genaue Implementierung der Tests geheim
⇒ Informationen nur in der Namensbeschreibung und den Details
- Sollen als grobe Hinweise auf Fehler dienen
⇒ Angabe/Manpages genau lesen ;-)

HINWEIS

Die Jobs/Tests der Gitlab-CI ...

- sind lediglich als zusätzliche Hilfestellung gedacht
- haben **KEINEN** Anspruch auf Vollständigkeit / Korrektheit
 - Erfolgreiche Pipeline kann 0 Punkte bedeuten, fehlgeschlagene Pipeline aber auch volle Punktzahl