



IT-Universitetet
i København

OBLIGATORISK OPGAVE # 2 I BOSC

Ooperativsystemer og C

Author:

Tom Mørk Christensen

Jonas Elbækgaard Jørgensen

ITU-mail:

TMCH@ITU.DK

JELB@ITU.DK

9. oktober 2014

Indhold

1	Delopgave 2	
	– Multitrådet FIFO buffer som kædet liste.	2
1.1	Implementationen, overordnet set	2
1.2	De specifikke løsninger	2
1.2.1	Tilføjelse af elementer	2
1.2.2	Fjernelse af elementer	2
1.2.3	Trådsiking	2
1.3	Fejl og mangler	2
1.4	Test	2
1.4.1	En anden tilgang til test	2

1 Delopgave 2

– Multitrådet FIFO buffer som kædet liste.

Denne delopgave har til formål at vise, at vi har forståelse for hvordan hægtedelister fungerer samt hvordan disse kan implementeres i C. Udover at demonstrere vores forståelse for hægtedelister tjener opgaven det formål at demonstrere hvordan nogle af risiciene ved flertrådet systemer kan forebygges.

1.1 Implementationen, overordnet set

Implementationen af vores løsning til denne delopgave begrænser sig til klassen `list.c`. I forbindelse med test af vores implementation har vi også ændret filen `main.c`, hvor vi har introduceret to nye funktioner `test()` og `add(void *param)`. Vi vil beskrive implementationen af disse funktioner samt deres formål i afsnit 1.4 på side 2.

I `list.c` kan vores implementation splittes i tre dele.

1. Tilføjelse af elementer til listen. Implementationen af denne funktionalitet begrænser sig til funktionen `list_add(List *l, Node *n)` og har til formål at hægte det nye element `n` bagerst på listen.
2. Fjernelse af elementer fra listen. Implementationen af denne funktionalitet begrænser sig til funktionen `list_remove(List *l)` og har til formål at fjerne og returnere de foreste element i den hægtede liste.
3. Trådsikring af listen, ved hjælp af gensidig udelukkelse. I modsætning til de to foregående dele er denne del spredt over flere af implementationens funktioner. Dette skyldes at mutexen skal oprettes sammen med listen, men låses og frigives i forbindelse med metodekald.

1.2 De specifikke løsninger

1.2.1 Tilføjelse af elementer

1.2.2 Fjernelse af elementer

1.2.3 Trådsikring

1.3 Fejl og mangler

Placeholder.

1.4 Test

Placeholder.

1.4.1 En anden tilgang til test

En anden mulig måde at teste hvorvidt vores hægtedeltiste understøtter ved brugen af funktionskaldet `pthread_mutex_trylock()` som returnere en fejl meddelelse hvis den givne mutex allerede låst. På denne måde kan man få de enkelte tråde til at printe en besked til terminalen når de er tvunget til at vente på at låsen på mutexen bliver frigivet.