

Portfolioprüfung – Werkstück A – Alternative 3

Aufgabe 1 Aufgabe

Entwickeln und implementieren Sie ein Programm zur Erzeugung und Visualisierung von Threads und Prozessen in Linux:

Entwickeln und implementieren Sie das geforderte System mit den entsprechenden Systemaufrufen oder Bibliotheksfunktionen. Nutzen Sie das Verzeichnis `/proc` um auf Prozessinformationen des Betriebssystems zuzugreifen und arbeiten Sie mit den von Linux zur Verfügung gestellten Filedeskriptoren `/proc/$pid/statm` u.a.

Innerhalb Ihrer Anwendung sollen die zentralen Informationen der erzeugten Prozesse (u.a. Prozess ID, Rechte, UID, GID, RAM-Ausnutzung, etc.) angezeigt und gespeichert werden. Weiter sollen Sie in Ihrem Programm mit der Systembibliothek `pthread.h` Threads erzeugen und mit Hilfe der durch die Bibliothek bereitgestellten Funktion sollen Threads gestartet und beendet werden können.

Außerdem sollen alle Funktionen für Threads (`pthread_join()`, `pthread_detach()`, `pthread_self()`, `pthread_equal()`) verwendet werden und die einzelnen Threads sollen auf geeignete Weise dargestellt werden. Entwickeln Sie eine geeignete Möglichkeit zur Visualisierung der Beziehungen der einzelnen Threads.

Zeigen Sie in Ihrem Programm die Unterschiede zwischen **User-Mode** und **Kernel-Mode** Threads auf. Benutzen Sie **Mutexe**, **Bedingungsvariablen** und **Semaphoren** zur Synchronisation der Threads und zeigen Sie die Eigenschaften der Synchronisationsmethoden auf.

Entwickeln und implementieren Sie Ihre Lösung als C/C++ Programm, als freie Software (Open Source) und verwenden Sie hierfür ein Code-Repository, z.B. bei GitHub.

Bearbeiten Sie die Aufgabe in Teams zu **4 Personen**.

Schreiben Sie eine aussagekräftige und ansehnliche Dokumentation (Umfang: **10 Seiten**) über Ihre Lösung.

Bereiten Sie einen Vortrag mit Präsentationsfolien und eine Live-Demonstration (Umfang: **15-20 Minuten**) vor. Demonstrieren Sie die Funktionalität der Lösung in der Übung.

Aufgabe 2 Anforderungen an das Programm

Das Programm muss folgende **fünf Grundfunktionen** implementieren:

1. Auswahl von Funktionen des Programms über Menü.
2. Ein-/ Ausgabe über CLI-Schnittstelle.
3. Schreiben der Ergebnisse in eine Datei.
4. Lesen der Ergebnisse aus der Datei.
5. Freigabe der Betriebssystemressourcen vor Beenden des Programms.

Weiter soll Ihr Programm folgende projektspezifische Anforderungen erfüllen:

- Ihre Anwendung soll eine Kommandozeilenanwendung sein.
- Der Quellcode soll durch Kommentare verständlich sein.
- Das Programm nutzt die Linux Systembibliothek `pthread.h`.
- Das Programm nutzt `fork()` und `exec()`.
- Das Programm visualisiert und speichert Prozessinformationen.
- Das Programm nutzt Mutexe, Bedingungsvariablen und Semaphore zur Synchronisation der Threads.
- Das Programm visualisiert die Beziehungen zwischen Prozessen und Threads.

Aufgabe 3 Literatur

- Foliensätze 4 und 6 der Vorlesung **Betriebssysteme und Rechnernetze** im SS2023
- **Linux-UNIX-Programmierung**, *Jürgen Wolf*, 2.Auflage, Rheinwerk Computing (2006), Kap. 10
- **Betriebssysteme kompakt**, *Christian Baun*, 2. Auflage, Springer Vieweg, S. 200-252
- **Betriebssysteme**, *Erich Ehses, Lutz Köhler, Petra Riemer, Horst Stenzel, Frank Victor*, 1. Auflage, Pearson (2005), S. 55-84
- **Betriebssysteme**, *Carsten Vogt*, 1. Auflage, Spektrum (2001), S. 109-127
- **Betriebssysteme**, *William Stallings*, 4. Auflage, Pearson (2003), S. 334-339