

Parallele Systeme SS2014

Übung 3

1/3 Vorbereitungsaufgaben (Bearbeitung vor dem Tutorium)

1. Installieren Sie auf Ihrem Notebook die gcc-Toolchain. Unter Windows kann beispielsweise auf die Umgebung „cygwin“ zurückgegriffen werden. Testen Sie die Installation kurz mit einem Aufruf von gcc und der Auflistung aller Parameter (auf welche Arten ist das möglich?) *[Keine Abgabe zu dieser Aufgabe notwendig]*

2/3 Übungsaufgaben (Bearbeitung im Tutorium)

2. Vorbereitung und Verwendung von fork
 1. Sehen Sie sich den zur Verfügung gestellten Quelltext „uebung03_1.c“ an. Gehen Sie jede Programmzeile durch und analysieren Sie, was und wozu jeweils gemacht wird. Konsultieren Sie ggf. jeweils die zugehörige „man page“ und/oder Beschreibungen im Internet. *[Keine Abgabe hierzu notwendig]*
 2. Tauschen Sie im Programm die Texte „[CHANGE ME]“ durch besser passende Texte (deutsch oder englisch) aus. Überlegen Sie, was bei Ausführung geschieht und welche Ausgaben Sie auf dem Terminal erwarten. Beschreiben Sie in Ihrer Abgabe kurz, welche Ausgabe Sie erwarten.
 3. Kompilieren Sie das Programm mit dem Aufruf „gcc -Wall -o uebung03_1 uebung03_1.c“ und führen Sie es aus. Was fällt Ihnen auf? Beschreiben Sie kurz, ob Sie ein unerwartetes Verhalten beobachten. Wenn ja, überlegen und beschreiben Sie kurz, was der Grund dafür ist. Führen Sie das Programm mehrmals aus.
 4. Nehmen Sie die vom Tutor beschriebene Änderung vor. Kompilieren und testen Sie das neue Programm. Beschreiben Sie kurz, welche Änderung es weshalb gibt.
 5. Versuchen Sie, die Lesbarkeit der Terminalausgabe mittels eines MUTEX zu verbessern (siehe die Verwendung der Variable `turn` im letzten Übungszettel). Verwenden Sie getrennte geschützte/kritische Bereiche für die verschiedenen Texte. Kompilieren und testen Sie erneut. Beschreiben Sie kurz, was Sie mit den Änderungen beobachten und was die Erklärung dafür ist. Prüfen Sie die Ausgabe der Programme `ps` (was bewirken die Optionen `a`, `u` und `x`?) und `top` während Ihrer Testläufe. Erklären Sie in Ihrer Beschreibung auch kurz, ob die Zuweisung „`myid = childcount;`“ im Widerspruch zu Ihrer Erklärung steht.
 6. (Optional:) Nehmen Sie Änderungen an dem Programmcode vor und untersuchen Sie, wie das Verhalten, insbesondere das Zusammenspiel der einzelnen Komponenten ist. Versuchen Sie, eine Lösung mit sogenannten „pipes“ zu realisieren. *[Keine Abgabe hierzu notwendig]*
 7. Beschreiben Sie kurz den Unterschied zwischen Prozessen und Threads.
3. Arbeiten mit POSIX-Threads
 1. Sehen Sie sich das zur Verfügung gestellte Programm „uebung03_2.c“ an. Analysieren Sie, was jeweils wozu gemacht wird. *[Keine Abgabe hierzu notwendig]*

2. Ändern Sie wieder die Texte „[CHANGE ME]“. Kompilieren und testen Sie das neue Programm (verwenden Sie jetzt den Aufruf „gcc -Wall -pthread -o uebung03_2 uebung03_2.c“). Vergleichen Sie sein Verhalten im Vergleich zu den verschiedenen Programmverhalten der letzten Aufgabe. Beschreiben Sie das Verhalten kurz.

3/3 Nachbereitungsaufgaben

4. Realisieren Sie die mittels MUTEX geschützte Terminalausgabe (siehe letzte Aufgabe) in dem Programm uebung03_2.c. Nutzen Sie hierfür noch keine speziellen MUTEX-Funktionen der pthread-Bibliothek. Kompilieren und testen Sie Ihre Implementierung. Führen Sie Ihr Programm mehrmals aus. Beschreiben Sie kurz Ihre Lösung und das beobachtete Verhalten. Das Programm ps kennt noch die Optionen „m“ und „s“, z.B. „ps axms“.
5. Implementieren Sie nun das MUTEX-Lock mit den pthread-Funktionen (pthread_mutex_init, pthread_mutex_lock, pthread_mutex_unlock, pthread_mutex_destroy). Kompilieren und testen Sie auch diese Variante (wieder mit mehreren Programmstarts). Beschreiben Sie kurz das Verhalten und Unterschiede zur letzten Implementierung. Was sind jeweils Vor- und Nachteile der beiden Varianten? Können Sie den Begriff „atomar“ für Ihre Erklärung verwenden?
6. (Optional:) Implementieren Sie ein Flag-basiertes MUTEX (boolesche Variable, kein „turn“) ohne die pthread-MUTEX-Funktionen. Versuchen Sie den Fehlerfall (mehr als ein Thread im kritischen Bereich) zu verursachen. *[Keine Abgabe hierzu notwendig]*
7. (Optional:) Implementieren Sie ein „Join“ des Elternthreads mit den Kindthreads. *[Keine Abgabe hierzu notwendig]*

Schicken Sie die Programme sowie die Antworten an Tim.Tiedemann@dfki.de