

Betriebssysteme

SS 2021

Dr.-Ing. Ahmad Rabie
ahmad.rabie@hs-ruhrwest.de

Aufgabe P2: Prozesse und Threads

Gegeben sind die folgenden Strukturen:

```
struct DeLorean
{
    double speed_in_kmh;
    double current_power_level_in_mega_watts;
};

struct IndexContainer
{
    unsigned int* cap_indices;
    unsigned int array_length;
};
```

Bearbeiten Sie mit Hilfe dieser Strukturen und der Lösung des letzten Praktikums die folgenden Aufgaben:

1. Erstellen Sie folgende globale Variablen:

```
struct DeLorean* delorean__;
struct FluxCapacitor** capacitors__;
```

Initialisieren Sie diese Variablen innerhalb der Main Funktion, d.h. instanzieren Sie die entsprechenden Strukturen. Hierbei sollen alle **FluxCapacitors** den Wert 10 im Attribut **value** erhalten, es sollen 121 FluxCapacitors angelegt werden.

2. Schreiben Sie eine Funktion `void* assembleDeLorean(void* indices_)`. Die Signatur dieser Funktion hat den Hintergrund, dass diese in einem POSIX Thread verwendet wird. Der Parameter `indices_` sei deshalb hinsichtlich seines Typs als `struct IndexContainer*` zu betrachten.

Die Aufgabe der Funktion ist:

- Auslesen der Elemente aus dem globalen **FluxCapacitor** Array, jedoch nur jener Elemente welche durch den **IndexContainer** indiziert werden.

- Addieren der `value` Attribute aller zuvor erwähnten Elemente.
 - Hinzufügen der obigen Summe zum Attribut `current_power_level_in_mega_watts` der globalen Variable `delorean__`.
3. Erstellen Sie zwei `IndexContainer`, einen mit 21 Elementen, einen weiteren mit 100 Elementen. Initialisieren Sie diese Container jeweils mit den Indizes 0 – 20 bzw. 21 – 120.
 4. Erstellen Sie nun zwei POSIX Threads, jeder dieser Threads soll jeweils die Funktion `assembleDeLorean` ausführen, die Parameter sind der erste bzw. zweite `IndexContainer`.
 5. Die Threads sollen sequentiell durchlaufen, d.h. der “zweite” Thread soll mit der Ausführung der Funktion `assembleDeLorean` erst starten nachdem der “erste” Thread fertig ist. Verwenden Sie hierfür einen Mutex innerhalb der genannten Funktion. Schlagen Sie hierzu die Funktionen `pthread_mutex_init`, `pthread_mutex_lock` sowie `pthread_mutex_unlock` nach.
 6. Nachdem beide Threads fertig sind, soll der Process forken. Der Child-Prozess soll prüfen ob das Attribut `current_power_level_in_mega_watts` der globalen Variable `delorean__` den Wert 1210 (1.21 GW) hat. Sofern dies zutrifft (und es sollte zutreffen), soll der Child-Prozess das Attribut `speed_in_kmh` der globalen Variable auf 144.1 (88 mp/h) setzen. Der Parent-Prozess soll warten bis der Child-Prozess beendet wurde.
 7. Achten Sie auf eine korrekte und vollständige Speicherfreigabe. Platzieren Sie sinnvolle `printf` Ausgaben.

Abgabe muss bis spätestens **04.05.2022, 23:59 Uhr** erfolgen.