

Aufgabe 1:

Schreiben Sie eine Producer/Consumer-Kommunikation über ein Shared Memory. Producer und Consumer sollen in verschiedenen Threads / Prozessen laufen.

Dafür wird im SHM der Platz für einen Container angelegt, der die Daten speichert und typisiert.

Der Producer soll diesen mit Daten füllen.

Der Consumer holt sich die Daten und zeigt sie an.

Der Handshake läuft über Semaphore im Container oder im SHM.

Frage: Wohin gehören sie?

1. Kopieren Sie sich die **CSharedMemory**- und **CBinarySemaphore**-Dateien (*.h, *.cpp) aus dem Framework ins Projekt. Ebenso **FWAssertion.h** und **Global.h**.
2. Schreiben Sie zunächst eine **CContainer**-Klasse. Diese soll zwei Membervariablen, **kmh** und **rpm** besitzen. Zusätzlich soll der Container zwei **Semaphoren** zur Synchronisation besitzen. Der zugriff auf die Daten soll über eine **setter**- bzw. **getter**-Methode erfolgen.
3. Implementieren Sie die **main()**-Funktion, die zunächst das SHM mit einen CContainer anlegt und dann den Producer- und den Consumer-Thread erzeugt.
4. Überprüfen Sie die Synchronisation zwischen Producer und Consumer indem Sie exemplarische Daten austauschen.

Hinweis: Hier werden die Libs **pthread** und **rt** benötigt und **#include <new>** muss inkludiert werden.

Viel Spaß!