

A2)

c.)

`std::unique_lock` kann alles was `std::mutex` auch kann, ist jedoch sicherer, da zB bei einer Exception im kritischen Abschnitt `unique_lock` den KA automatisch wieder entsperrt, genauso wenn das entsperren (`unlock`) vergessen wurde. `std::unique_lock` ist also eine verbesserte Variante von `std::mutex` die noch zusätzliche Funktionen mitbringt.

d.)

Bei `BusyWaiting` wird solange über eine Schleife iteriert, bis die Bedingung auf die gewartet wird erfüllt ist (verbraucht CPU-Zeit). Bei Verwendung einer `condition_variable` wird auch gewartet, jedoch nicht mit einer Schleife und die CPU muss dafür auch nicht die ganze Zeit arbeiten. Stattdessen wird der Thread an der Stelle „schlafengelegt“ und erst dann geweckt, wenn ein anderer Thread diesen aufweckt.

Im Gegensatz zu `BusyWaiting` ist eine `condition_variable` deutlich ressourcensparender/effizienter, da keine CPU-Zeit sinnlos „verschwendet“ wird. Zudem blockiert `BusyWaiting` im kritischen Abschnitt den Ablauf, da dann kein anderer Thread diesen nun noch betreten kann.