Como visto em aula, o problema do produtor/consumidor com buffer limitado consiste em um buffer de dada capacidade onde um produtor grava dados sequencialmente e, de forma concorrente, o consumidor retira os dados na mesma ordem em que foram gravados pelo produtor (FIFO). Em sala de aula foi visto uma solução para este problema utilizando monitores.

Desenvolva uma solução em C/C++ utilizando as operações mutexbegin/mutexend e block/wakeup(p) para implementar as operações produzir(int dado) e consumir() de modo a evitar inconsistências no buffer e, ao mesmo tempo, evitar que uma thread fique bloqueada desnecessariamente.

A solução deve atender aos seguintes requisitos:

* A solução deve ter pelo menos 1 produtor e 1 consumidor, cada um implementado por uma thread;
* Se o buffer estiver cheio, a operação produzir deve bloquear o produtor até que a operação possa ser concluída corretamente;
* Se o buffer estiver vazio, a operação consumir deve bloquear o consumidor até que a operação possa ser concluída corretamente;
* Se o buffer não estiver cheio nem vazio, tanto produtor quanto consumidor devem poder acessá-lo;
* O buffer deve ser implementado através de um array.

Utilize a implementação das funções block e wakeup disponíveis no slide de aula na forma como estão. Você NÃO deve modificar a implementação das mesmas.

A sua implementação pode ser realizada utilizando pthread\_t ou std::thread. Da mesma forma, pode-se usar pthread\_mutex\_t ou std::mutex.