Lab Exercise

- **Make producer and consumer working at the same time**
  - The producer puts the number of LOOP in the queue
  - The consumer gets the number of LOOP in the queue
  - If the queue is full, producer can't put it and if queue is empty, consumer can't get it
    - You should use a synchronization mechanism using CV

Lab Exercise

- **Make `put_data` and `get_data` function.**
  - `void put_data(queue_t* queue, int data);`
  - `int get_data(queue_t* queue);`

- **All produced data must be consumed**
  - You should use a synchronization mechanism using CV
  - As many data as the number of LOOP must be generated and consumed.
  - The number of consecutive lines with “get/put” does not have to be equal to QSIZE.

```
put data 0 to queue
put data 1 to queue
put data 2 to queue
put data 3 to queue
put data 4 to queue
get data 0 from queue
get data 1 from queue
put data 5 to queue
put data 6 to queue
get data 2 from queue
get data 3 from queue
get data 4 from queue
get data 5 from queue
get data 6 from queue
put data 7 to queue
put data 8 to queue
put data 9 to queue
put data 10 to queue
put data 11 to queue
get data 7 from queue
get data 8 from queue
get data 9 from queue
get data 10 from queue
get data 11 from queue
put data 12 to queue
put data 13 to queue
put data 14 to queue
put data 15 to queue
put data 16 to queue
get data 12 from queue
get data 13 from queue
get data 14 from queue
get data 15 from queue
get data 16 from queue
```

Exercise Submission

- **Submit your source code and Makefile**
  - The ***make*** command should generate a ***w15*** executable.
  - via iCampus
  - Bundle ***source code*** and ***Makefile*** with tar command
    - *tar.gz* format

    $ tar cvzf *[student_id].tar.gz week15*
  - We'll grade your submission with **make**
    - If compilation fails, your points for this exercise will be zero