N Producers/ M Consumers

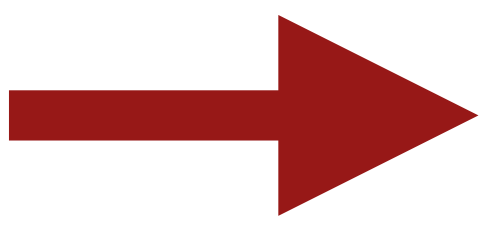
```
std::vector<std::future<void>> consumers(M);
std::vector<std::future<void>> producers(N);

int i = 0;
for (auto& p : producers)
  p = std::async(std::launch::async, push, std::ref(q), i++);

for (auto& c : consumers)
  c = std::async(std::launch::async, pop, std::ref(q));

for (auto& c : consumers)
  c.wait();

for (auto& p : producers)
  p.wait();
```

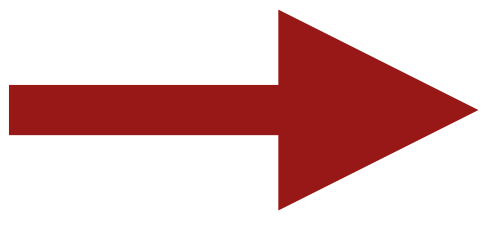
Producer

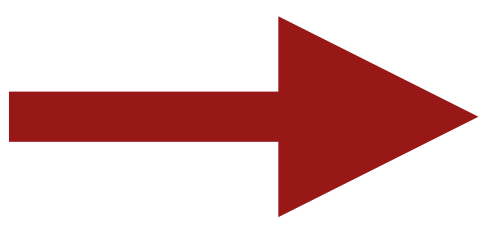
```
void push(Q& q, unsigned id)
{
    for(unsigned i=id*LOOP_SIZE; i<(id+1)*LOOP_SIZE; ++i)
        q.push(i);
}
```

Consumer

```
void pop(Q& q)
{
    unsigned i;
    while(true)
    {
        q.pop(i);
    }
}
```







Producer

```
void push(Q& q, unsigned id)
{
    for(unsigned i=id*LOOP_SIZE; i<(id+1)*LOOP_SIZE; ++i)
        q.push(i);
}
```

Consumer

```
void pop(Q& q)
{
    unsigned i;
    while(true)
    {
        q.pop(i);
    }
}
```

N Producers/ M Consumers

```
std::vector<std::future<void>> consumers(M);
std::vector<std::future<void>> producers(N);

int i = 0;
for (auto& p : producers)
  p = std::async(std::launch::async, push, std::ref(q), i++);

for (auto& c : consumers)
  c = std::async(std::launch::async, pop, std::ref(q));

for (auto& c : consumers)
  c.wait();

for (auto& p : producers)
  p.wait();
```

```
Processes: 284 total, 3 running, 6 stuck, 275 sleeping, 1329 threads
15:37:55
Load Avg: 1.52, 1.58, 1.49  CPU usage: 7.0% user, 31.40% sys, 61.59% idle   SharedLibs: 1196K resident,
0B data, 0B linkedit.
MemRegions: 90828 total, 2462M resident, 65M private, 620M shared. PhysMem: 8116M used (1303M wired), 73M
unused.
VM: 724G vsize, 1026M framework vsize, 8785612(0) swapins, 9281977(0) swapouts. Networks: packets:
26748134/33G in, 15538508/3923M out.
Disks: 2499813/95G read, 3051851/126G written.

PID    COMMAND      %CPU  TIME     #TH   #WQ  #PORT MEM    PURG   CMPRS  PGRP  PPID  STATE    BOOSTS
%CPU_ME %CPU_OTHRS UID  FAULTS
26591  top          3.4   00:00.89 1/1   0    19    3084K  0B     0B     26591 25987 running  *0[1]
0.00000 0.00000    0    39637+
26590  a.out        99.8  00:22.29 5/1   0    13    368K   0B     0B     26590 25842 running  *0[1]
0.00000 0.00000    501  342
26585  Python       0.0   00:01.75 32    0    53    37M    0B     0B     26585 26575 sleeping *0[1]
0.00000 0.00000    501  31977
26575  Vim          0.0   00:01.60 31    0    71    22M    0B     0B     26575 5829  sleeping *0[4]
0.00000 0.00000    501  13019
```