#include <stdio.h>

#include <pthread.h>

#include <semaphore.h>

#define B 5

int buffer[B], in = 0, out = 0;

sem\_t empty, full;

pthread\_mutex\_t mutex;

void \*producer(void \*param) {

for (int i = 0; i < 10; i++) {

sem\_wait(&empty);

pthread\_mutex\_lock(&mutex);

buffer[in] = i; in = (in + 1) % B;

printf("Produced %d\n", i);

pthread\_mutex\_unlock(&mutex);

sem\_post(&full);

}

return NULL;

}

void \*consumer(void \*param) {

for (int i = 0; i < 10; i++) {

sem\_wait(&full);

pthread\_mutex\_lock(&mutex);

int item = buffer[out]; out = (out + 1) % B;

printf("Consumed %d\n", item);

pthread\_mutex\_unlock(&mutex);

sem\_post(&empty);

}

return NULL;

}

int main() {

pthread\_t prod, cons;

sem\_init(&empty, 0, B); sem\_init(&full, 0, 0);

pthread\_mutex\_init(&mutex, NULL);

pthread\_create(&prod, NULL, producer, NULL);

pthread\_create(&cons, NULL, consumer, NULL);

pthread\_join(prod, NULL); pthread\_join(cons, NULL);

sem\_destroy(&empty); sem\_destroy(&full); pthread\_mutex\_destroy(&mutex);

return 0;

}

