

```
1  /* prod-cons.c
2  *   code without process synchronization
3  */
4
5  #include <pthread.h>
6  #include <stdio.h>
7  #include <stdlib.h>
8  #include <unistd.h>
9
10 #define BUFSIZE 2
11 #define MAXDELAY 3000000 // usec (3 sec)
12
13 #define rand2() ((double)rand() / RAND_MAX)
14
15 struct buffer {
16     int buf[BUFSIZE]; // buffer
17     int in; // next free position
18     int out; // next full position
19     int count; // number of items in the buffer
20 };
21
22 void initbuf(struct buffer *p);
23 void *producer(void *arg);
24 void *consumer(void *arg);
25 void putmsg(struct buffer *p, int msg);
26 int getmsg(struct buffer *p);
27
28 int main()
29 {
30     struct buffer bbuf;
31     pthread_t tid1, tid2;
32
33     initbuf(&bbuf);
34     pthread_create(&tid1, NULL, producer, &bbuf);
35     pthread_create(&tid2, NULL, consumer, &bbuf);
36     pthread_join(tid1, NULL);
37     pthread_join(tid2, NULL);
38     return 0;
39 }
40
41 void initbuf(struct buffer *p)
42 {
43     p->in = p->out = p->count = 0;
44 }
45
46 void *producer(void *arg)
47 {
48     struct buffer *p = (struct buffer *)arg;
49     int msg=0;
50
51     while (1) {
52         usleep(rand2()*MAXDELAY);
53         msg++;
54         printf("Producer inserts message %d\n", msg);
55         putmsg(p,msg);
56     }
57 }
58
59 void *consumer(void *arg)
60 {
61     struct buffer *p = (struct buffer *)arg;
62     int msg;
63
```

```
64     while (1) {
65         usleep(rand2()*MAXDELAY);
66         msg = getmsg(p);
67         printf("Consumer removes message %d\n", msg);
68     }
69 }
70
71 void putmsg(struct buffer *p, int msg)
72 {
73     while (p->count == BUFSIZE)
74         ;
75     p->count++;
76     p->buf[p->in] = msg;
77     p->in = (p->in + 1) % BUFSIZE;
78
79     printf("putmsg: buffer size = %d\n", p->count);
80 }
81
82 int getmsg(struct buffer *p)
83 {
84     int msg;
85
86     while (p->count == 0)
87         ;
88     p->count--;
89     msg = p->buf[p->out];
90     p->out = (p->out + 1) % BUFSIZE;
91
92     printf("> getmsg: buffer size = %d\n", p->count);
93
94     return msg;
95 }
96
```