

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <pthread.h>
5
6 // thread deadlock example
7
8 int n=1000000;
9
10 pthread_mutex_t mutex0;
11 pthread_mutex_t mutex1;
12
13 int num0=0;
14 int num1=0;
15
16 void* run0(void * arg){
17     for(int i=0;i<n;i++){
18         pthread_mutex_lock(&mutex1);
19         sleep(1);
20         pthread_mutex_lock(&mutex0);
21
22         num0++;
23         num1--;
24         printf("num1 -->%d\n",num1);
25         pthread_mutex_unlock(&mutex0);
26         pthread_mutex_unlock(&mutex1);
27     }
28     return NULL;
29 }
30
31 void* run1(void * arg){
32     for(int i=0;i<n;i++){
33         pthread_mutex_lock(&mutex1); // change mutex for deadlock yes/no
34         sleep(1);
35         pthread_mutex_lock(&mutex0);
36         num0--;
37         num1++;
38         printf("num0 -->%d\n",num0);
39         pthread_mutex_unlock(&mutex0);
40         pthread_mutex_unlock(&mutex1);
41     }
42     return NULL;
43 }
44
45 int main(int arg, char* argv[])
46 {
47     pthread_t thid0,thid1;
48     pthread_mutex_init(&mutex0,NULL);
49     pthread_mutex_init(&mutex1,NULL);
50
51     int ris = pthread_create(&thid0, NULL, &run0, NULL);
52     if (ris){
53         printf("errore creazione thread\n");
54         exit(-1);
55     }
56     ris = pthread_create(&thid1, NULL, &run1, NULL);
57     if (ris){
58         printf("errore creazione thread\n");
59         exit(-1);
60     }
```

```
60     }  
61     pthread_join(thid0, NULL);  
62     pthread_join(thid1, NULL);  
63     pthread_mutex_destroy(&mutex0);  
64     pthread_mutex_destroy(&mutex1);  
65 }  
66
```