#include<stdio.h>

#include<pthread.h>

#include<unistd.h>

int count;

int flag[2]={0,0};

int turn=0;

void\* process(void\* arg)

{

int i=\*(int\*)arg;

int other=1-i;

for(count=0;count<5;count++)

{

flag[i]==1;

turn=other;

while(flag[other]==1 && turn==other);

printf("Process %d is in critical section\n",i);

sleep(1);

flag[i]=0;

printf("Process %d is in remainder section\n",i);

sleep(1);

}

return NULL;

}

int main()

{

pthread\_t p1,p2;

int id1=0,id2=1;

pthread\_create(&p1,NULL,process,&id1);

pthread\_create(&p2,NULL,process,&id2);

pthread\_join(p1,NULL);

pthread\_join(p2,NULL);

return 0;

}

/\*

gcc peterson.c -pthread

./a.out

Process 1 is in critical section

Process 0 is in critical section

Process 1 is in remainder section

Process 0 is in remainder section

Process 1 is in critical section

Process 0 is in critical section

Process 1 is in remainder section

Process 0 is in remainder section

Process 1 is in critical section

Process 0 is in critical section

Process 1 is in remainder section

Process 0 is in remainder section

\*/