

## Exercise

Write a program that creates two threads.

Thread 1 must print the even numbers (from 2 to 10) with the message: th1:number. Thread 2 must print the odd numbers (from 1 to 9) with the message: th2:number. The program must print the numbers on the screen in increasing order. For achieving this, the executions of the two threads must alternate and the format of the messages on the screen must be:

```
th1: 1
th2: 2
th1: 3
th2: 4
th1: 5
th2: 6
th1: 7
th2: 8
th1: 9
th2: 10
```

The synchronization must be done with mutex. It is not allowed the use of sleep or of any other synchronization mechanism.

## Solution

```
/*
THREADS
Creates 2 threads alternating execution.
José Manuel Pérez Lobato
*/

#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>

pthread_attr_t attr;
pthread_t idth[10];
pthread_mutex_t mtx;
pthread_cond_t varcond;
int turn=1;

void *thread1(void *num) {
    int cont=1;

    while (cont <=10){
        pthread_mutex_lock (&mtx);
        while (turn!=1) pthread_cond_wait(&varcond, &mtx);
        printf ("th1:%d\n", cont);
        cont=cont+2;
        turn = 2;
        pthread_cond_signal(&varcond);
        pthread_mutex_unlock (&mtx);
    }
}
```

Tema: 7

Este material se comparte con licencia Creative Commons Reconocimiento no Comercial-Compartida-Igual 3.0. España.



```
}
    pthread_exit(0);
}
void *thread2(void *num) {
    int cont=2;

    while (cont <=10){
        pthread_mutex_lock (&mtx);
        while (turn!=2) pthread_cond_wait(&varcond, &mtx);
        printf ("th2:%d\n", cont);
        cont=cont+2;
        turn = 1;
        pthread_cond_signal(&varcond);
        pthread_mutex_unlock (&mtx);
    }
    pthread_exit(0);
}

int main(){
    int i;

    pthread_mutex_init (&mtx, NULL);
    pthread_attr_init(&attr);

    pthread_create(&idth[0], &attr, thread1, &i);
    pthread_create(&idth[1], &attr, thread2, &i);
    // waiting the termination of threads
    for (i=0; i<2; i++)
        pthread_join(idth[i], NULL);
    return(0);
}
```