

[Tema 2-Sección crítica y exclusión mútua](#)

Es la parte del código que accede a un recurso compartido con otro proceso o thread. Hay que controlar esto para que no se produzcan inconsistencias en la información. Por ejemplo:

```
#include <thread.h>

int i=0;

int increment(void *args){
    int v=i; //Sección crítica
    v++; //Sección crítica
    i=v; //Sección crítica

    return 0;
}

int main(){
    thrd_t t1, t2;

    thrd_create(&t1, increment, NULL);
    thrd_create(&t2, increment, NULL);
    thrd_join(&t1, NULL);
    thrd_join(&t2, NULL);

    return 0;
}
```

Es la parte del código donde se toman recursos compartidos, como la variable i. El thread 2 entra en la función antes de que se modifique el valor en el thread 1. Por tanto, la operación i++ solo se hace una vez. Este problema se conoce como **lost update**.