

Considere os seguintes processos:

```
int mutex==1;
proctype P1()      ||      proctype P2()
{
  bool c1=false;   {
do
  bool c2=false;
if
  do
  if
  ::mutex==1->mutex=0;      ::mutex==1->mutex=0;
                          c2=true;
  c1=true;
  ::mutex==0 & c1          ::mutex==0 & c2
  -> c1=false;            -> c2=false;
                          mutex=2;
  mutex=2;
  ::true-> skip;          ::true-> skip;
fi;                       fi;
od                         od
\                           }
}
```

- 1) Supondo que as variáveis $c1$ e $c2$ definem zonas críticas, indique se é possível que ambos os processos estejam na zona crítica. Como pode especificar a propriedade a verificar em LTL?

Sim é possível:

P1 corre até $\text{if } \text{mutex}==1 \rightarrow$ e o escalonador muda o controlo para P2

P2 corre até $\text{if } \text{mutex}==1 \rightarrow$

$G(\sim c1 \text{ or } \sim c2)$

- 2) Resolva o problema anterior recorrendo a semáforos. Que tipo de justiça é necessária nas ações de $\text{acquire}()$ e $\text{release}()$? Especifique em LTL as fórmulas que especificam esta justiça!

```

semaphore mutex==1;
proctype P1()      ||
{
bool c1=false;
do
if
true->
    12::mutex.acquire
    13::c1=true;
13::c1 ->
    14::mutex.release
    15::
::true-> skip;
fi;
od
^

```

```

proctype P2()
{
bool c2=false;
do
if
::true->mutex.acquire;
    c2=true;
::c2 ->
    mutex.release
::true-> skip;
fi;
od
}

```

Acquire justiça forte $GF(P1@12) \Rightarrow GF(P1@13)$
Release justiça fraca $FG(P1@14) \Rightarrow GF(P1@15)$