

NOMBRE->CHRISTOFER FABIÁN CHÁVEZ CARAZAS  
TEMA-> A

## 1. Código

```
MODULE main
DEFINE
  n1 := 1;
  n2 := 2;
VAR
  x: {1,2};
  P1: process persona(P2.b_my,x,n1);
  P2: process persona(P1.b_my,x,n2);
LTLSPEC !(F(P1.state = l3 & P2.state = l3));

MODULE persona(b_other,x,number)
VAR
  b_my: boolean;
  state : {l1,l2,l3};
ASSIGN
  init(b_my) := TRUE;
  init(state) := l1;
  next(state) := case
    state = l1 : l2;
    state = l2 & (x = number | b_other = FALSE) : l3;
    state = l3 : l1;
    TRUE: state;
  esac;

  next(b_my) := case
    state = l1 : TRUE;
    state = l3 : FALSE;
    TRUE: b_my;
  esac;

  next(x) := case
    state = l1 & number = 1: 2;
    state = l1 & number = 2: 1;
    TRUE: x;
  esac;
```

## 2. Sentencia

Lo que nos pide el problema es probar que dos personas no puedan ingresar a la cuenta bancaria al mismo tiempo. Esto se podría traducir en la siguiente sentencia LTL:

$$!(F(P1.state = l3 \ \& \ P2.state = l3))$$

Si un objeto persona esta en el estado l3, entonces esta dentro de la cuenta bancaria: “En un futuro dos personas no podran estar en la cuenta bancaria al mismo tiempo”

## 3. Test

```
NuSMV > check_ltlspec
-- specification !( F (P1.state = l3 & P2.state = l3)) is true
NuSMV > |
```