

## Example — effects of non-confluence and non-termination

```
fleck@tornado [101]% cafeobj
-- CafeOBJ system Version 1.4.2 --
CafeOBJ> in nonconfl
processing input : ./nonconfl.mod
-- defining module! NONCONFL....._..* done.
CafeOBJ> show NONCONFL
module! NONCONFL
{
  imports {
    protecting (BOOL)
  }
  signature {
    [ Split ]
    op a : -> Split
    op b : -> Split
    op f : Split Split -> Split
  }
}
```

```

axioms {
  var X : Split
  var Y : Split
  eq f(a,Y) = Y .
  eq f(X,b) = X .
}
}
CafeOBJ> select NONCONFL
NONCONFL> red f(a,b) .
-- reduce in NONCONFL : f(a,b)
b : Split
(0.000 sec for parse, 1 rewrites(0.030 sec), 1 matches)
NONCONFL> red a == b .
-- reduce in NONCONFL : a == b
false : Bool
(0.000 sec for parse, 1 rewrites(0.000 sec), 1 matches)
NONCONFL> red f(a,b) == a .
-- reduce in NONCONFL : f(a,b) == a
false : Bool
(0.010 sec for parse, 2 rewrites(0.000 sec), 2 matches)

```

```

NONCONFL> in nonterm
processing input : ./nonterm.mod
-- defining module! NONTERM....._* done.
NONCONFL> show NONTERM
module! NONTERM
{
  imports {
    protecting (BOOL)
  }
  signature {
    [ Looping ]
    op f : Looping Looping -> Looping { strat: (0 1 2) }
    op a : -> Looping
    op b : -> Looping
  }
  axioms {
    var X : Looping
    var Y : Looping
    eq f(X,Y) = f(Y,X) .
  }
}

```

```
NONCONFL> select NONTERM  
NONTERM> red f(a,b) .  
-- reduce in NONTERM : f(a,b)  
*** - Lisp stack overflow. RESET  
[CHAOS]>  
fleck@tornado [102]% exit
```