Exercise 2

CafeOBJ Team of JAIST

Prove confluence

- Prove the following NAT+' confluent

```
mod! BASIC-NAT{
  [Zero NzNat < Nat]
  op 0 : -> Zero
  op s_ : Nat -> NzNat
}
```

```
mod! NAT+' {
  pr(BASIC-NAT)
  op _+_ : Nat Nat -> Nat
  vars M N : Nat
  eq N + 0 = N .
  eq M + s N = s(M + N) .
  eq 0 + N = N .
  eq s M + N = s(M + N) .
}
```

Find a counter-example

- Termination is a necessary condition for the following critical pairs theorem
  - Theorem (Knuth and Bendix 1970): If SP is terminating and all critical pairs are joinable, then SP is confluent
- Find an example whose all critical pairs are joinable but which is not confluent
  - Hint: It is not terminating

Reverse of reverse

- Prove rev (rev L) = L .

```
mod! BASIC-LIST(X :: TRIV) {  [Empty NeList < List]
  op nil : -> Empty
  op _::_ : Elt List -> NeList
}
```

```
mod! LIST-@-assoc { pr(BASIC-LIST)
  op _@_ : List List -> List {assoc}
  var E : Elt .  vars L1 L2 : List .
  eq nil @ L1 = L1 .
  eq (E :: L1) @ L2 = E :: (L1 @ L2) .
}
```

```
mod! LIST-rev { pr(LIST-@-assoc)
  op rev _ : List -> List
  var E : Elt .  var L : List .
  eq rev nil = nil .
  eq rev (E :: L) = (rev L) @ (E :: nil) .
}
```

Exercises

1. Prove NAT+' confluent
2. Find an example whose all critical pairs are joinable, but which is not confluent
3. Prove rev (rev L) = L