

```

library libraryCompte

from Basic/Numbers get Rat
from Basic/SimpleDatatypes get Boolean
from Basic/CharactersAndStrings get String

spec COMPTE =
    Rat
and Boolean
and String

then sort Compte
    pred estCloture : Compte
    ops
        creer : Compte;
        crediter : Compte * Rat ->? Compte;
        debiter : Compte * Rat ->? Compte;
        cloturer : Compte ->? Compte;
        titulaire : Compte -> String;
        solde : Compte -> Rat;

forall c1, c2 : Compte; s1 : Rat; p1: String
    . def crediter(c1, s1) <=> not estCloture(c1)
    . def debiter(c1, s1) <=> not estCloture(c1)
    . def cloturer(c1) <=> not estCloture(c1) /\ solde(c1)=0

```

```
. not estCloture(creer)
. solde(creer) = 0

. estCloture(cloturer(c1))
. solde(cloturer(c1)) = 0
. titulaire(cloturer(c1)) = titulaire(c1)

. solde(crediter(c1, s1)) = solde(c1) + s1
. titulaire(crediter(c1, s1)) = titulaire(c1)

. solde(debiter(c1, s1)) = solde(c1) - s1
. titulaire(debiter(c1, s1)) = titulaire(c1)
```

end