```
data ℕ : Set where
  zero : ℕ
  suc : ℕ → ℕ
{-# BUILTIN NATURAL ℕ #-}

infix 6 _+_
infix 4 _≡_

_+_ : ℕ → ℕ → ℕ
zero + n = n
(suc m) + n = suc (m + n)
{-# BUILTIN NATPLUS _+_ #-}

data _≡_ {A : Set} (x : A) : A → Set where
  refl : x ≡ x

1+1≡2 : 1 + 1 ≡ 2
1+1≡2 = refl
```