可換的な引き算

令和 4 年 6 月 23 日

目録

```
1  module Op where
```

Implementation of Subtraction

```
2  data R = R Double deriving Show
```

Defining subtraction as following, hence satisfying $a - b = b - a$.

```
3  instance Num R where
4      (-) (R a) (R b) = case a>=b of
5                  True -> R $ a-b
6                  False -> R $ b-a
7      fromInteger a = R $ abs x
8                  where x = fromIntegral a::Double
9      signum (R a)
10              | a==0 = 0
11              | otherwise = 1
12      abs a = a
13      (*) _ _ = notImplemented
```

Now let's try to implement (+) normally

```
14      (+) (R a) (R b) = R $ a+b
```

But such implementation yields a wierd behaviour where (let a>b) then
$b + (a - b) = a$
$a + (a - b) \neq b$

Extended Definition of Real Number Space

```
15    data R1 = R1 R Bool deriving Show
16    data R2 = R2 R Double deriving Show
```

```
17    instance Num R1 where
18        (-) a b = notImplemented
```

```
19        abs (R1 a _) = R1 a False
20        fromInteger a
21                    | a>=0 = R1 (R x) False
22                    | a <0 = R1 (R x) True
23                        where x = fromIntegral a::Double
24        signum (R1 (R a) d)
25                    | a==0 = 0
26                    | d==True = -1
27                    | d==False = 1
28        (+) a b = notImplemented
29        (*) _ _ = notImplemented
```