observation pattern

pattern

```
tail cycle Z m = cycle m m
```

argument patterns

copattern clause

copattern

```
tail tail fib = zipWith (+) fib (tail fib)
```

observation pattern   nested copattern

argument patterns

```
prepend (x :: xs) s = x :: (prepend xs s)
```

pattern clause