```
insert :: Ord a => a -> [a] -> [a]
insert x [] = [x]                        [0]
insert x (y:ys) | x < y    = x:y:ys      [1]
                | otherwise = y:(insert x ys)  [2]

isort :: Ord a => [a] -> [a]
isort [] = []                            [4]
isort (x:xs) = insert x (isort xs)       [3]
```

isort [4,3,2]

```
[3] = insert 4 (isort [3,2])
[3] = insert 4 (insert 3 (isort [2]))
[3] = insert 4 (insert 3 (insert 2 (isort [])))
[4] = insert 4 (insert 3 (insert 2 []))
[0] = insert 4 (insert 3 [2])
[2] = insert 4 (2: insert 3 [] )
[0] = insert 4 [2,3]
[2] = 2:(insert 4 [3])
[2] = 2:(3: insert 4 [])
[0] = 2: 3:[4]
    = [2,3,4]
```

```
qsort []= []     [0]
qsort (x:xs) = qsort [y | y<-xs, y<x] ++ [y | y<-xs, y==x] ++ [x] ++ qsort [y | y<-xs, y>x]  [1]
```

qsort [4,3,2]

```
[1]= qsort [3,2] ++ [] ++ [4] ++ qsort []
[0]= qsort [3,2] ++ [] ++ [4] ++ []
   = qsort [3,2] ++ [4]

[1]= qsort [2] ++ [] ++ [3] ++ qsort [] ++ [4]
[0]= qsort [2] ++ [] ++ [3] ++ [] ++ [4]
   = qsort [2] ++ [3,4]

[1] = qsort [] ++ [] ++ [2] ++ qsort [] ++ [3,4]
[0] = [] ++ [] ++ [2] ++ [] ++ [3,4]
    = [2,3,4]
```