

```
-- @QNone stands for submatrix which elements are zeroes
```

```
data QTree a = QNone  
             | QVal a  
             | QNode (QTree a) (QTree a) (QTree a) (QTree a)
```

Purely functional sparse
data representation

```
main = ...  
      let new_members = eWiseAdd gt prob neighbor_max  
          iset' = eWiseAdd lor iset new_members in  
      ...
```

Excerpt from Luby's
maximal independent set
implementation

```
--gets fused into  
main = ...
```

Matrix element-wise
addition operation

Sparse
matrices

After distillation the
excerpt becomes an implicit
function that takes three
input matrices instead of
two

```
let iset' = case iset of  
    ... -> case neighbor_max of  
    ... -> case prob of ...  
-- @new_members has been eliminated
```

Since these operations are connected via
'new_members' we can make a 'transitive
closure' and fuse them into one, thus
reducing the need for construction and
destruction of 'new_members'