

```

1 local _,the,SYM = require"lib", require"the", require"sym"
2 local fmt,per,upx,push,sort = _._fmt,_.per,_.upx,_.push,_.sort
3 local ent,o,oo = _._ent,_.o, _._oo
4 local class,OBJ = _._class, _._OBJ
5
6 local BIN=class("BIN",OBJ)
7 function BIN:new(at,name, lo,hi,ys)
8   self.at, self.name = at or 0, name or ""
9   self.lo, self.hi, self.ys = lo, hi or lo, ys or SYM() end
10
11 function BIN:_tostring()
12   local x,lo,hi,big = self.name, self.lo, self.hi, math.huge
13   if lo == hi then return fmt("%s==%s",x, lo)
14   elseif hi == big then return fmt("%s>=%s",x, lo)
15   elseif lo == -big then return fmt("%s<=%s",x, hi)
16   else return fmt("%s<=%s<%s",lo,x,hi) end end
17
18 function BIN:select(row)
19   local x, lo, hi = row[self.at], self.lo, self.hi
20   return x=="?" or lo == hi and lo == x or lo <= x and x < hi end
21
22 function BIN:add(x,y)
23   if x<self.lo then self.lo = x end
24   if x>self.hi then self.hi = x end
25   self.ys:add(y) end
26
27 function BIN.mergeSameDivs(b4,after)
28   local merged = b4.ys:merged(after.ys)
29   if merged then
30     return BIN(b4.at, b4.name, b4.lo, after.hi, merged) end end
31
32 function BIN.mergeNext(b4,after)
33   if b4.hi == after.lo then
34     return BIN(b4.at, b4.name, b4.lo, after.hi, b4.ys:merge(after.ys)) end end
35
36 return BIN
37
38 -- 23 45 {:left 100 :right 45} n=145

```