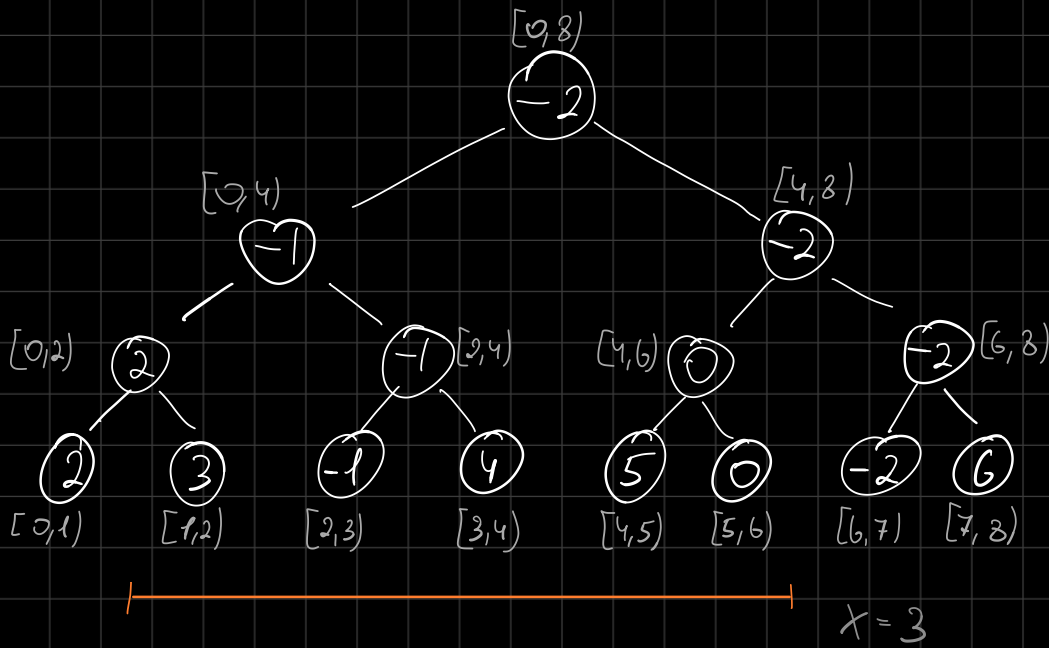


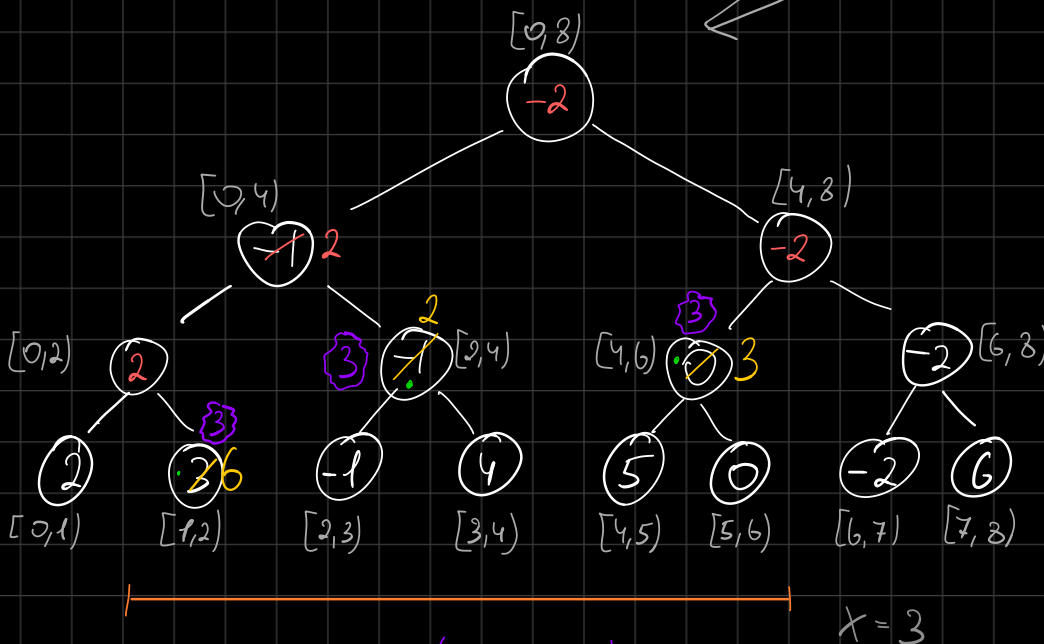
# Дерево отрезков

• get  $l$   $r$  //  $\min(a_l, a_{l+1}, \dots, a_r)$

• add  $l$   $r$   $x$  //  $a_l += x, a_{l+1} += x, \dots, a_r += x$



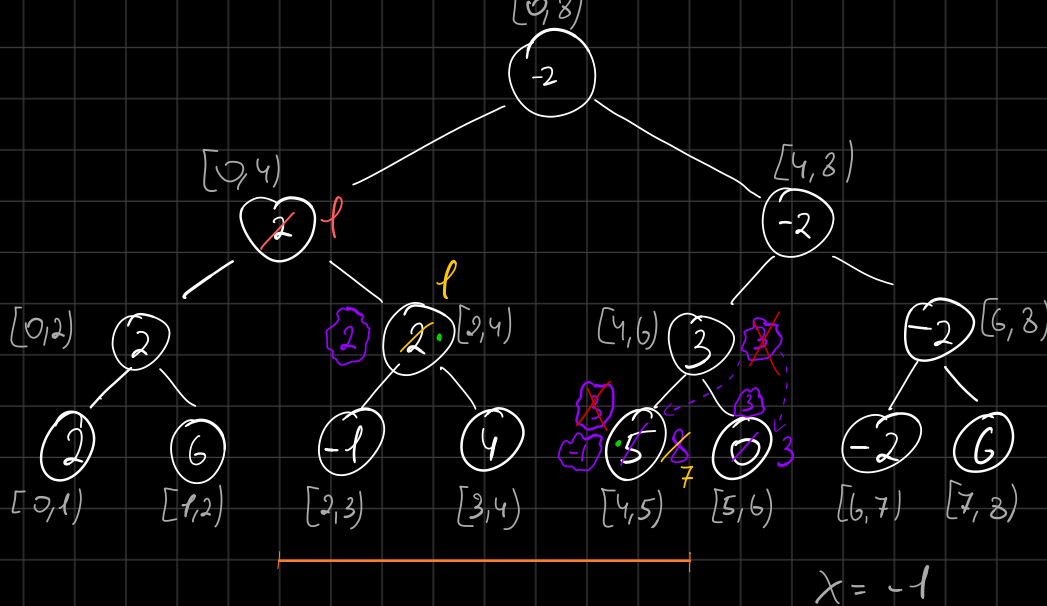
исп. Ленивые/отложенные операции



■ - хотим к детям прибавить  $x$  ( $\text{add}[v]$ )

■ - меняем отрезок

■ - прибавл. к вершине  $x$



Условия:

1) На входе значение корректное

2)  $add[v]$  хранит, сколько прибавить ко всем детям (не считая текущ. вершины)

$add(v, l, r, ql, qr, x)$

if ( $qr \leq l$  ||  $r \leq ql$ ):

return

if ( $ql \leq l$  &&  $r \leq qr$ ):

$tree[v] += x$

$add[v] += x$

return

$m = (l + r) / 2$

push(v)

$add(2v+1, l, m, ql, qr, x)$

$add(2v+2, m, r, ql, qr, x)$

$tree[v] = \min(tree[2v+1], tree[2v+2])$

push(v)

if ( $add[v] == 0$ ): return

$tree[2v+1] += add[v]$

$tree[2v+2] += add[v]$

$add[2v+1] += add[v]$

$add[2v+2] += add[v]$

$add[v] = 0$

get(v, l, r, ql, qr):

if (qr ≤ l || r ≤ ql):

return +∞

if (ql ≤ l && r ≤ qr):

return tree[v]

push(v)

m = (l+r)/2

return min(get(2v+1, l, m, ql, qr), get(2v+2, m, r, ql, qr))

• get l r // sum( $a_l, a_{l+1}, \dots, a_r$ )

• add l r x

get(v, l, r, ql, qr):

if (qr ≤ l || r ≤ ql):

return 0

if (ql ≤ l && r ≤ qr):

return tree[v]

push(v)

m = (l+r)/2

return sum(get(2v+1, l, m, ql, qr), get(2v+2, m, r, ql, qr))

$add(v, \overset{le}{l}, \overset{rg}{r}, ql, qr, x)$

if ( $qr \leq l$  ||  $r \leq ql$ ):

return

if ( $ql \leq l$  &&  $r \leq qr$ ):

$tree[v] += x \cdot (r - l)$

$add[v] += x$

return

$m = (l + r) / 2$

$push(v, l, m, r)$

$add(2v+1, l, m, ql, qr, x)$

$add(2v+2, m, r, ql, qr, x)$

$tree[v] = \min(tree[2v+1], tree[2v+2])$

• get  $l$   $r$  //  $\sum(q_l, q_{l+1}, \dots, q_{r-1})$

• add  $l$   $r$   $x$  //  $q_l = x, q_{l+1} = x, \dots, q_{r-1} = x$

$get(v, l, r, ql, qr)$ :

if ( $qr \leq l$  ||  $r \leq ql$ ):

return 0

if ( $ql \leq l$  &&  $r \leq qr$ ):

return  $tree[v]$

$push(v)$

$m = (l + r) / 2$

return  $\sum(get(2v+1, l, m, ql, qr), get(2v+2, m, r, ql, qr))$

$push(v, l, m, r)$

if ( $add[v] == 0$ ): return

$tree[2v+1] += add[v] \cdot (m - l)$

$tree[2v+2] += add[v] \cdot (r - m)$

$add[2v+1] += add[v]$

$add[2v+2] += add[v]$

$add[v] = 0$

$add(v, \overset{le}{l}, \overset{rg}{r}, ql, qr, x)$

if ( $qr \leq l$  ||  $r \leq ql$ ):

return

if ( $ql \leq l$  &&  $r \leq qr$ ):

$tree[v] = x \cdot (r - l)$

$add[v] += x$

return

$m = (l + r) / 2$

$push(v, l, m, r)$

$add(2v+1, l, m, ql, qr, x)$

$add(2v+2, m, r, ql, qr, x)$

$tree[v] = \min(tree[2v+1], tree[2v+2])$

$push(v, l, m, r)$

if ( $add[v] == 0$ ): return

$tree[2v+1] = add[v] \cdot (m - l)$

$tree[2v+2] = add[v] \cdot (r - m)$

$add[2v+1] += add[v]$

$add[2v+2] += add[v]$

$add[v] = 0$

set x  $\Leftrightarrow$  set y  
set y

$add\ l\ r\ x\ d \ //\ q_l += x, q_{l+1} += x+d, q_{l+2} += x+2d, \dots, q_{r-1} += x+(r-l)d$

$\left. \begin{array}{l} x_1\ d_1 \rightarrow x_1\ x_1+d_1\ x_1+2d_1\ \dots \\ x_2\ d_2 \rightarrow x_2\ x_2+d_2\ x_2+2d_2\ \dots \end{array} \right\} \Rightarrow (x_1+x_2)\ (d_1+d_2)$

