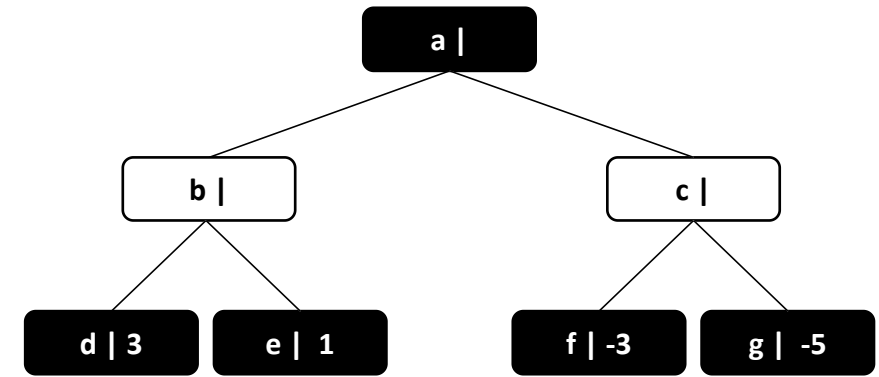


Start: AlphaBetaMax(a, $-\infty$, ∞)

Node: a α : $-\infty$ β : ∞

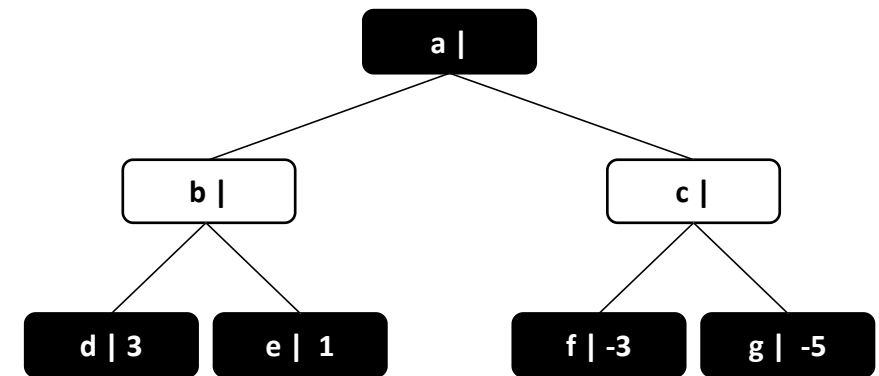


Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)
IF isLeafNode(a) is true THEN Return MinMaxValue(a) //false
childrenNodes = getChildren(Node)
WHILE childrenNodes is NOT empty
 α = max(α , AlphaBetaMin(first(childrenNodes), α , β))
 IF $\alpha \geq \beta$ THEN Return β
 childrenNodes = rest(childrenNodes)
Return α

Node: a α : $-\infty$ β : ∞

isLeafNode(a): false



Start: AlphaBetaMax(**a**, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(**a**) is true THEN Return MinMaxValue(**a**)

childrenNodes = getChildren(**a**)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: **a**

α : $-\infty$

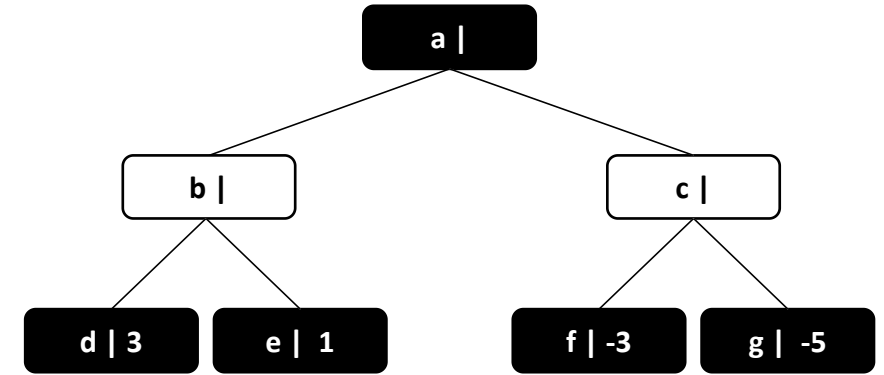
β : ∞

isLeafNode(**a**): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

$\alpha: -\infty$

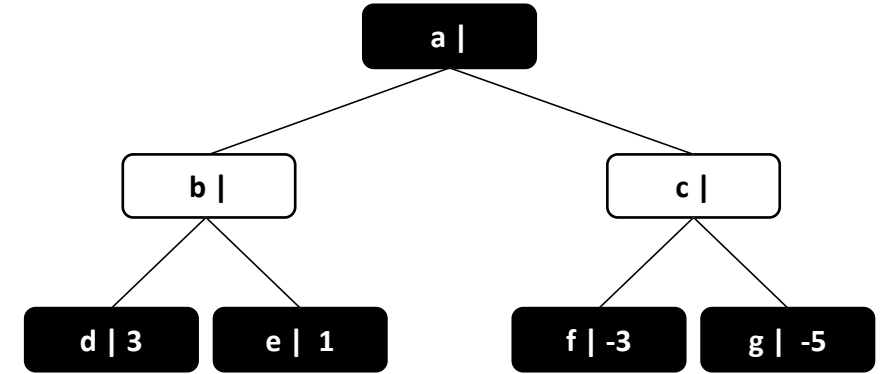
$\beta: \infty$

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

Node: a

$\alpha: -\infty$

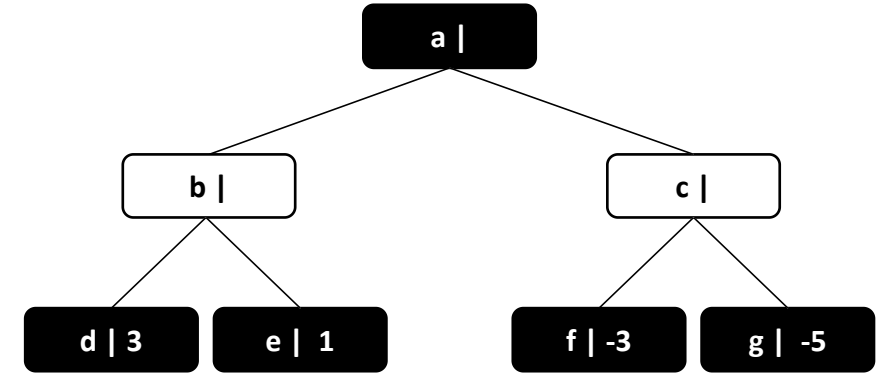
$\beta: \infty$

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first(childrenNodes)}, \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: a

α : $-\infty$

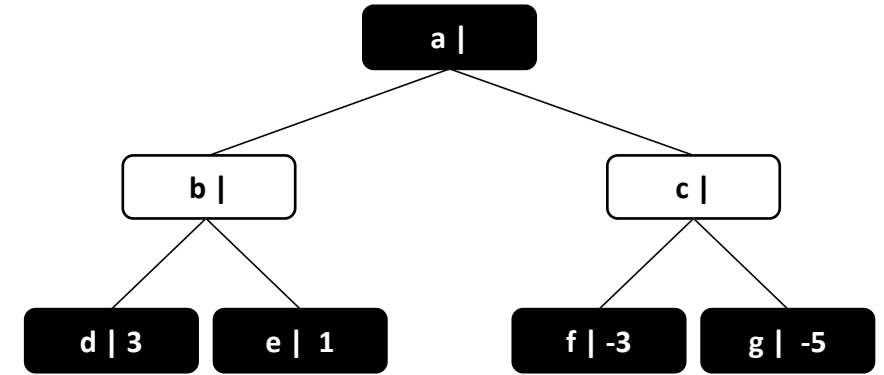
β : ∞

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b) //false

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

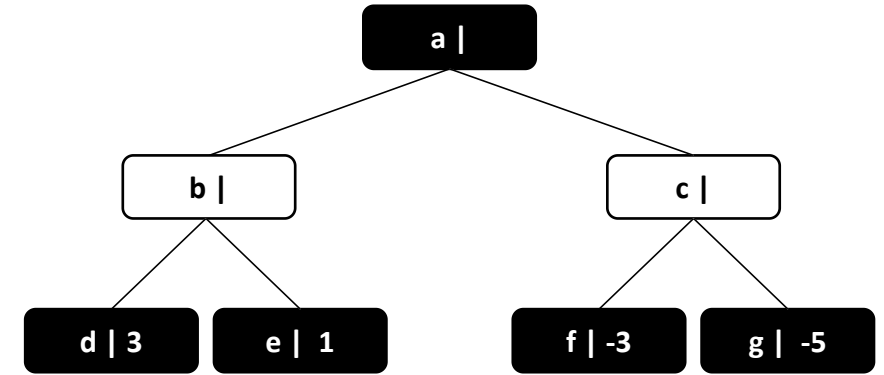
Return β

Node: b

α : $-\infty$

β : ∞

isLeafNode(b): false



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

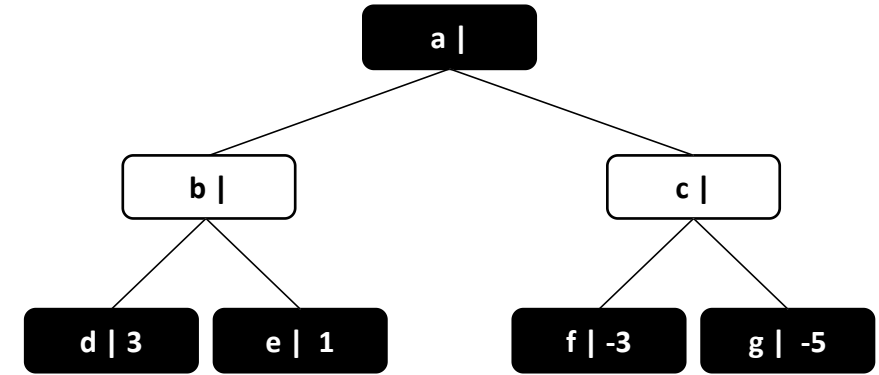
Node: b

α : $-\infty$

β : ∞

isLeafNode(b): false

childrenNodes: d, e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

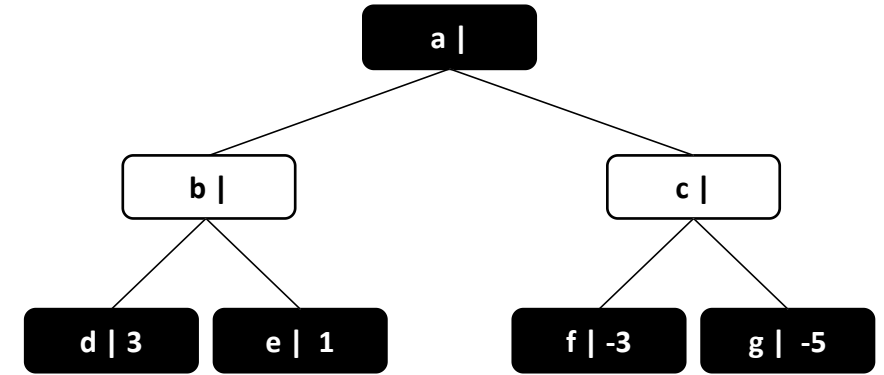
Node: b

α : $-\infty$

β : ∞

isLeafNode(b): false

childrenNodes: d, e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$ ←

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

Node: b

α : $-\infty$

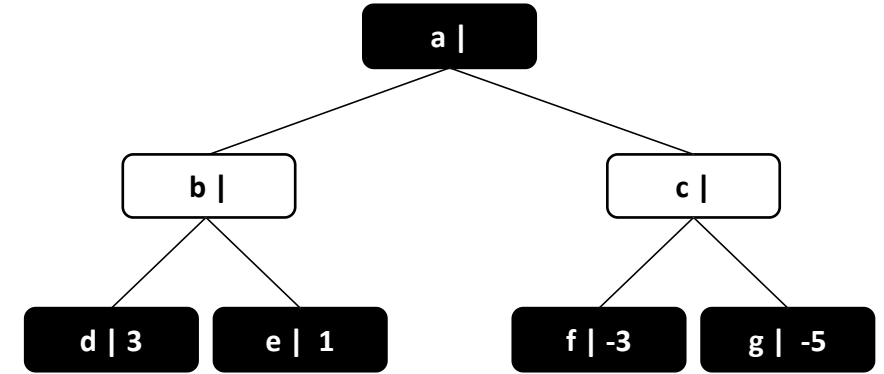
β : ∞

isLeafNode(b): false

childrenNodes: d, e

first(childrenNodes): d

rest(childrenNodes): e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: b

α : $-\infty$

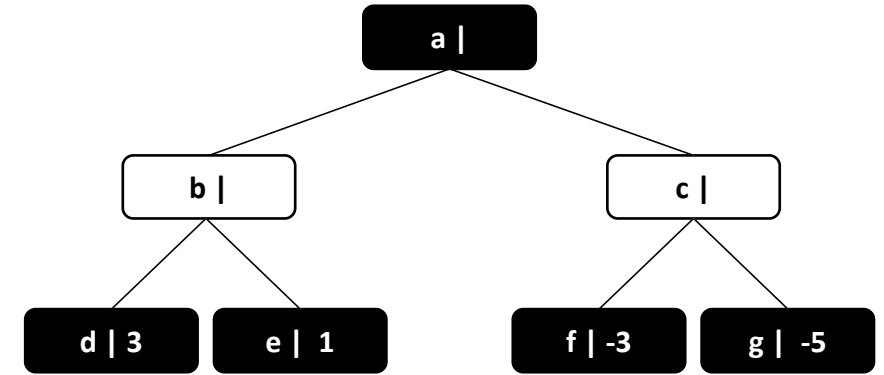
β : ∞

isLeafNode(b): false

childrenNodes: d, e

first(childrenNodes): d

rest(childrenNodes): e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

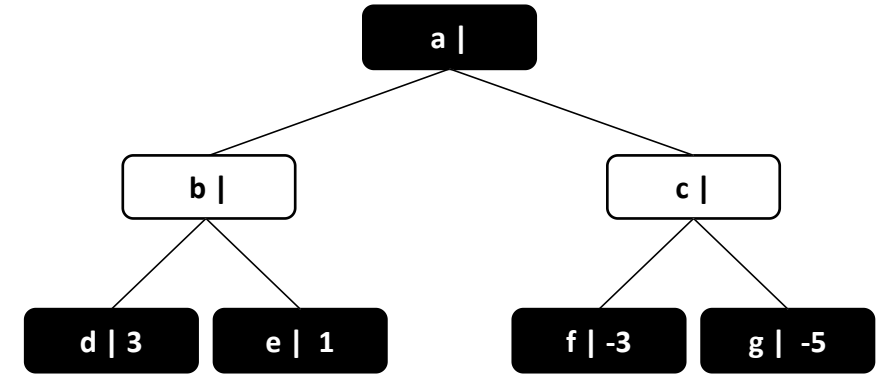
childrenNodes = rest(childrenNodes)

Return α

Node: d

$\alpha: -\infty$

$\beta: \infty$



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(d) is true THEN Return MinMaxValue(d) //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

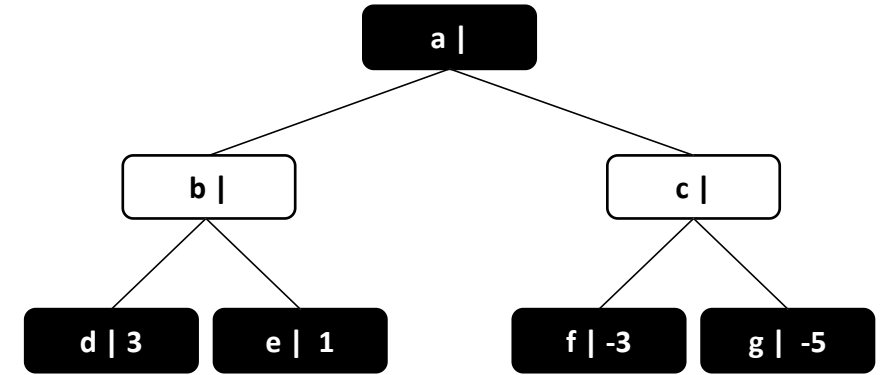
Return α

Node: d

$\alpha: -\infty$

$\beta: \infty$

isLeafNode(d): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(d) is true THEN Return 3 //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

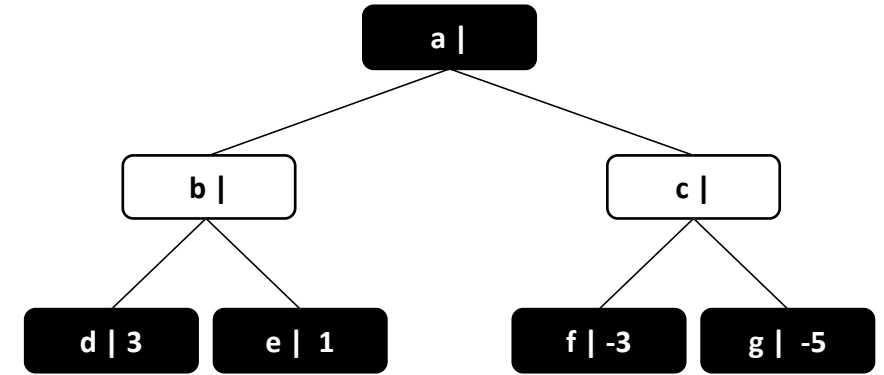
Return α

Node: d

$\alpha: -\infty$

$\beta: \infty$

isLeafNode(d): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, \text{AlphaBetaMax}(d, -\infty, \infty))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

3
returned

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(d) is true THEN Return 3 //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

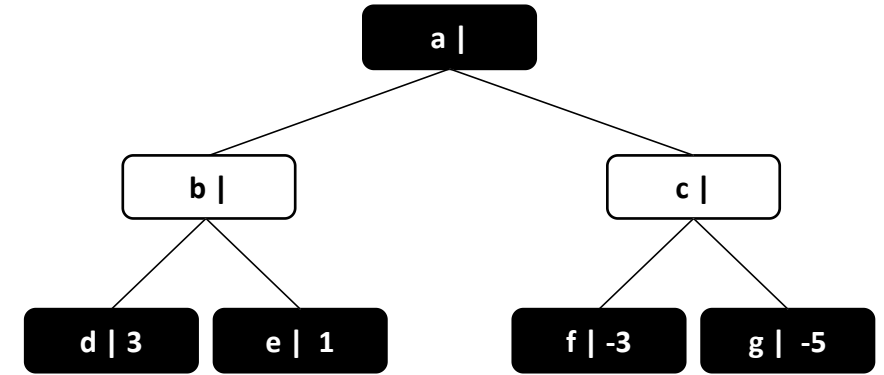
Return α

Node: d

$\alpha: -\infty$

$\beta: \infty$

isLeafNode(d): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\infty, 3)$ // we can calculate and get 3

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

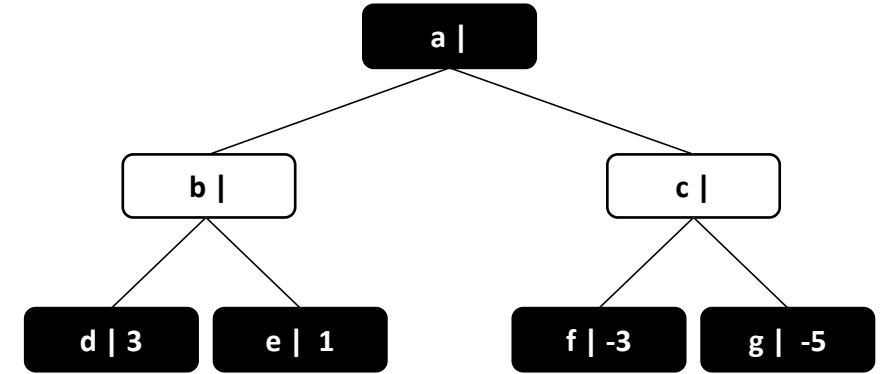
b: 3

isLeafNode(b): false

childrenNodes: d, e

first(childrenNodes): d

rest(childrenNodes): e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = 3$ // we can calculate and get 3

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

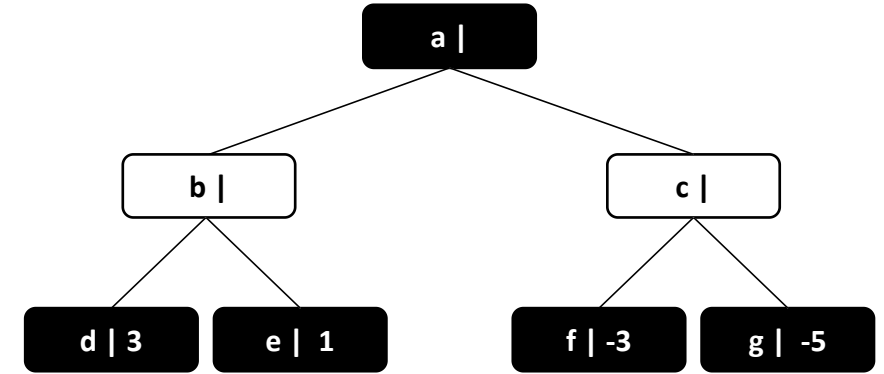
b: 3

isLeafNode(b): false

childrenNodes: d, e

first(childrenNodes): d

rest(childrenNodes): e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = 3$

IF $\beta \leq \alpha$ THEN Return α // $\beta \leq \alpha$ is false

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

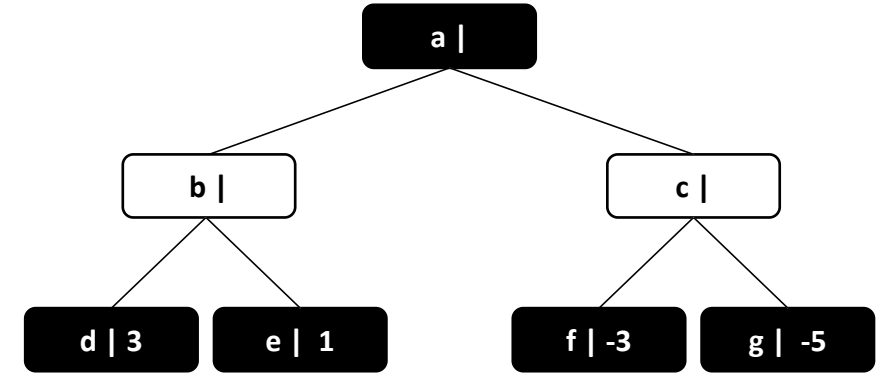
b: 3

isLeafNode(b): false

childrenNodes: d, e

first(childrenNodes): d

rest(childrenNodes): e



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = 3$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

$\alpha: -\infty$

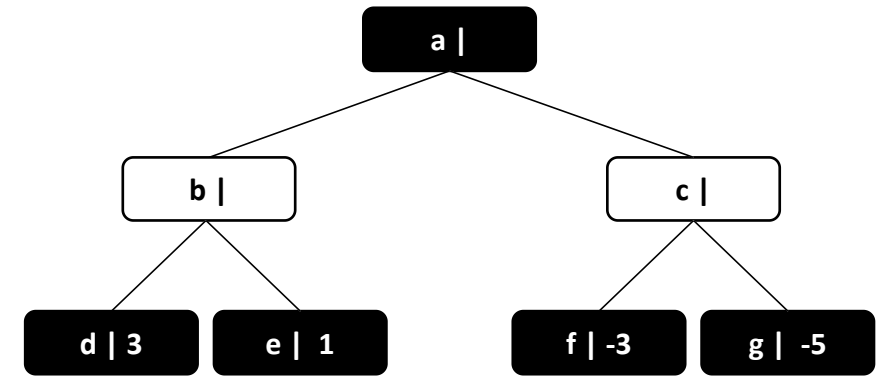
b: 3

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

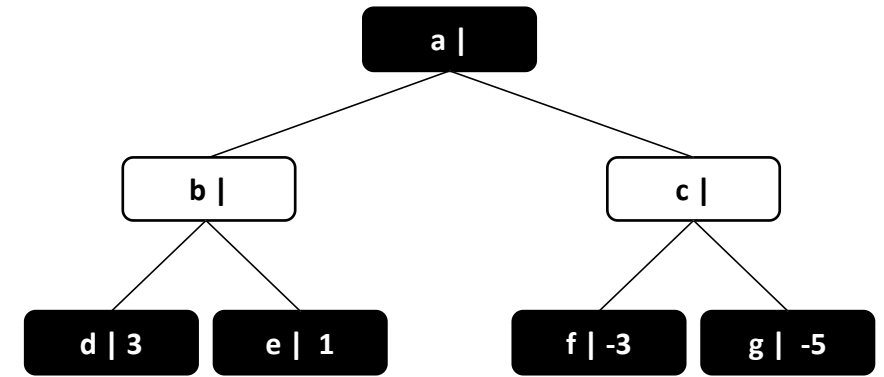
b: 3

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

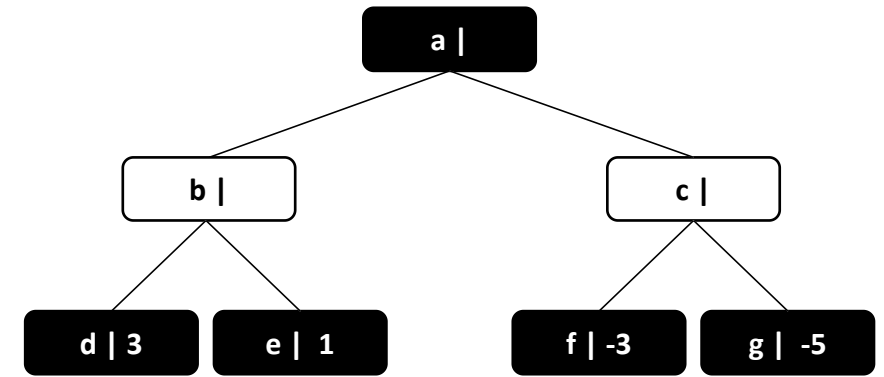
b: 3

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(e, -\infty, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMax Value(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: b

α : $-\infty$

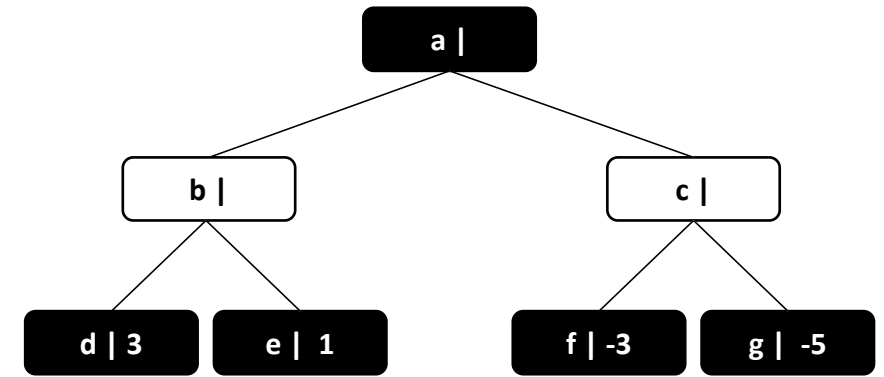
b: 3

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(e, -\infty, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(e) is true THEN Return MinMaxValue(e) //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

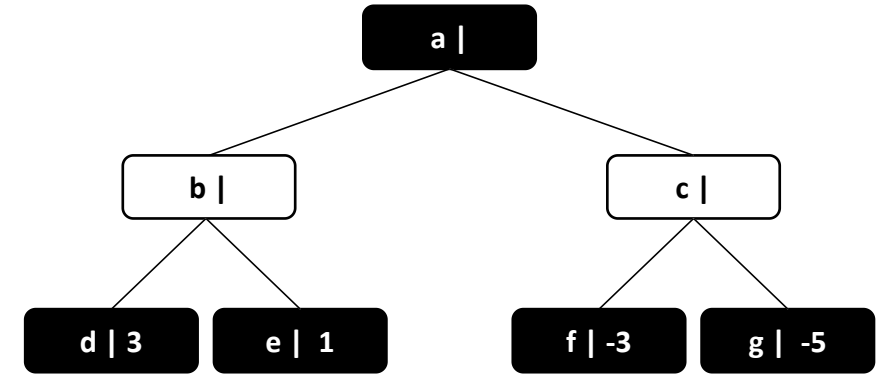
childrenNodes = rest(childrenNodes)

Return α

Node: e
isLeafNode(e): true

α : $-\infty$

β : 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(e, -\infty, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(e) is true THEN Return 1 //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

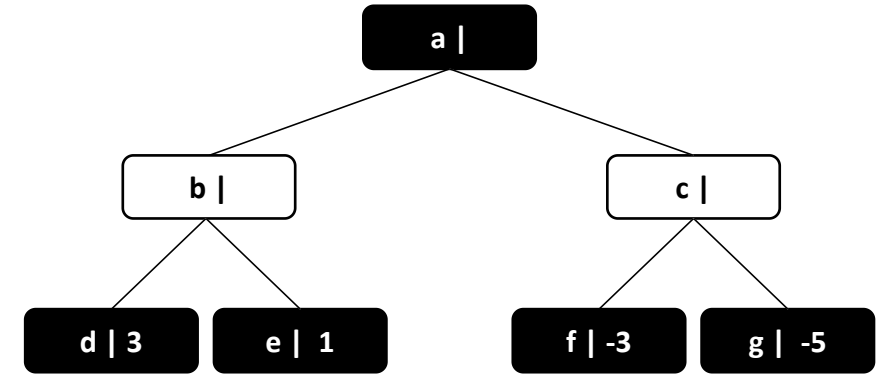
childrenNodes = rest(childrenNodes)

Return α

Node: e
isLeafNode(e): true

α : $-\infty$

β : 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(e, -\infty, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

1
returned

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(e) is true THEN Return 1 //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

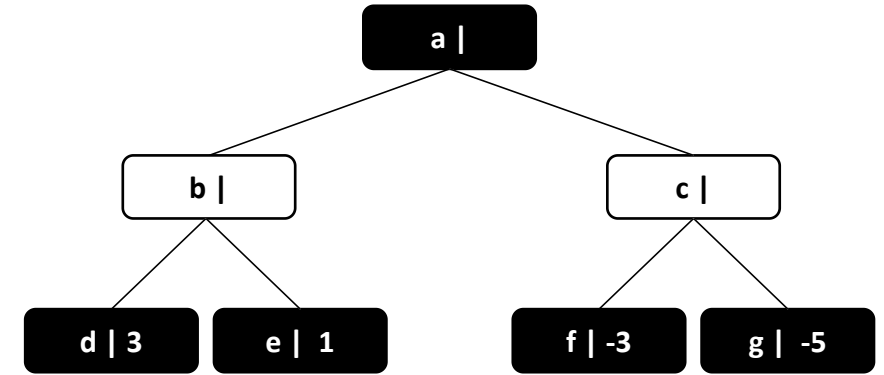
childrenNodes = rest(childrenNodes)

Return α

Node: e
isLeafNode(e): true

α : $-\infty$

β : 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, 1)$ // we can calculate and get 1

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

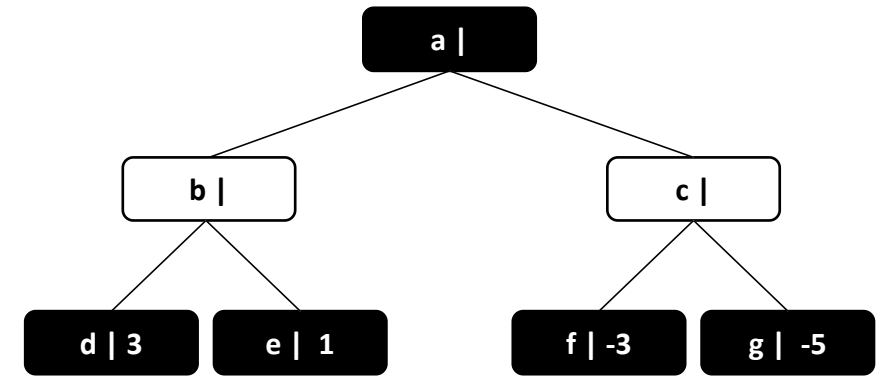
b: 3

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = 1$

IF $\beta \leq \alpha$ THEN Return α // $\beta \leq \alpha$ is false

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

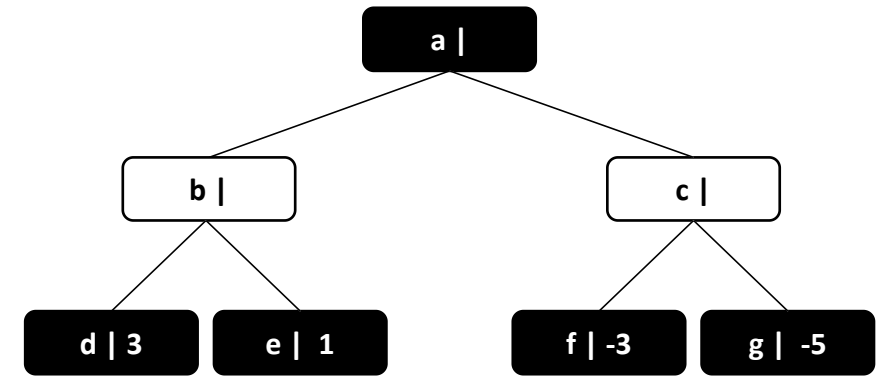
b: 1

isLeafNode(b): false

childrenNodes: e

first(childrenNodes): e

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = 1$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

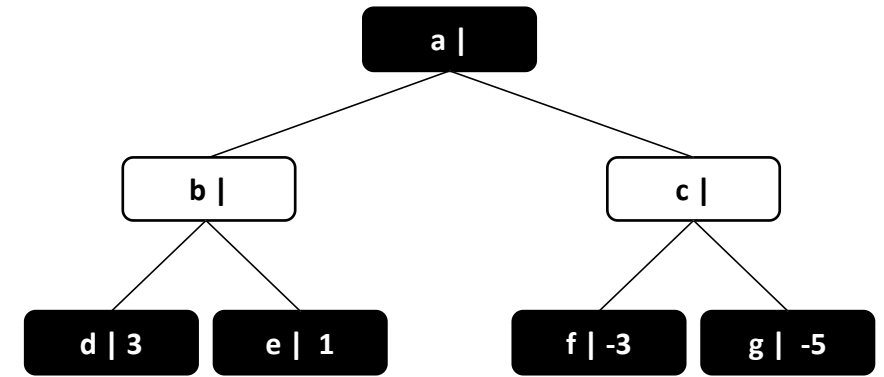
b: 1

isLeafNode(b): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //empty! leave the loop

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: b

α : $-\infty$

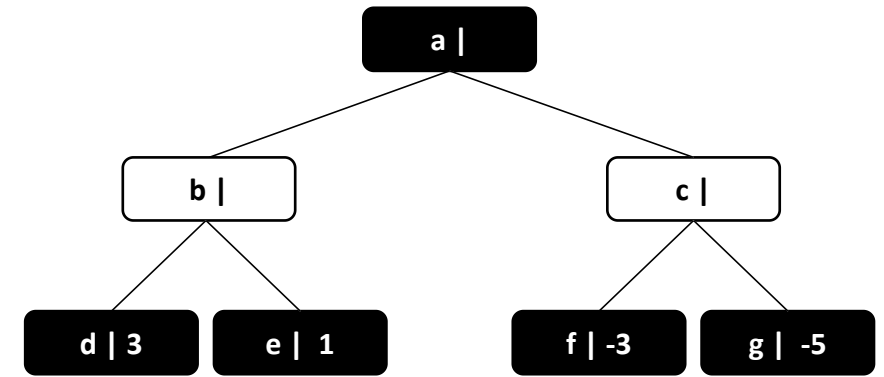
b: 1

isLeafNode(b): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //empty! leave the loop

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β // time to go back

Node: b

α : $-\infty$

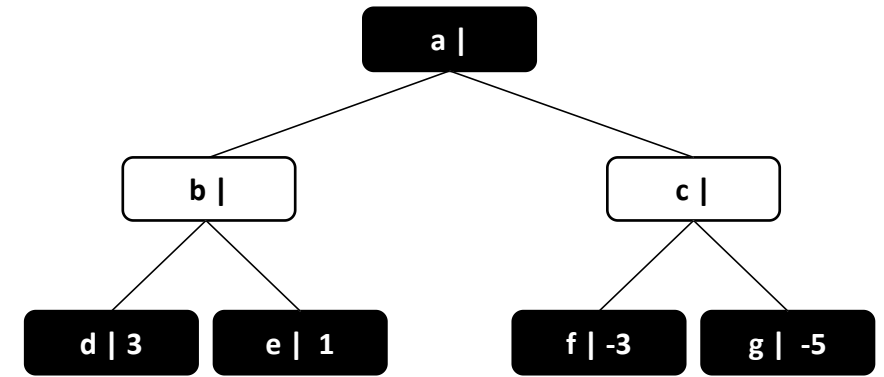
b: 1

isLeafNode(b): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$ ←

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //empty! leave the loop

$\beta = \min(3, 1)$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β // time to go back and return beta

Node: b

α : $-\infty$

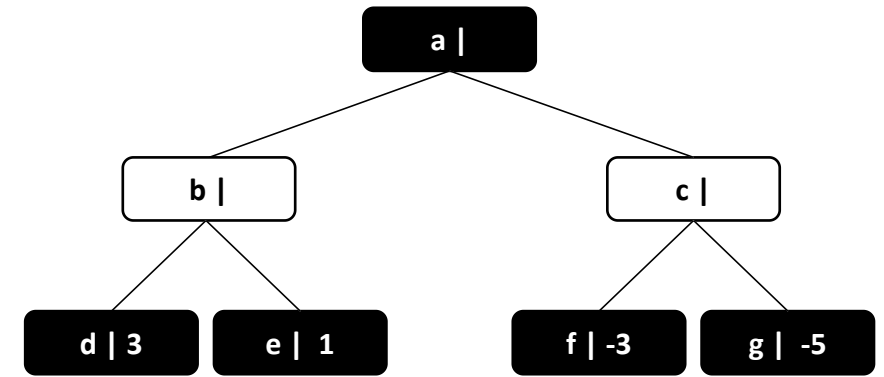
b: 1

isLeafNode(b): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, \text{AlphaBetaMin}(b, -\infty, \infty))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

1

returned

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(b) is true THEN Return MinMaxValue(b)

childrenNodes = getChildren(b)

WHILE childrenNodes is NOT empty //empty! leave the loop

$\beta = \min(3, 1)$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β // time to go back and return beta

Node: b

α : $-\infty$

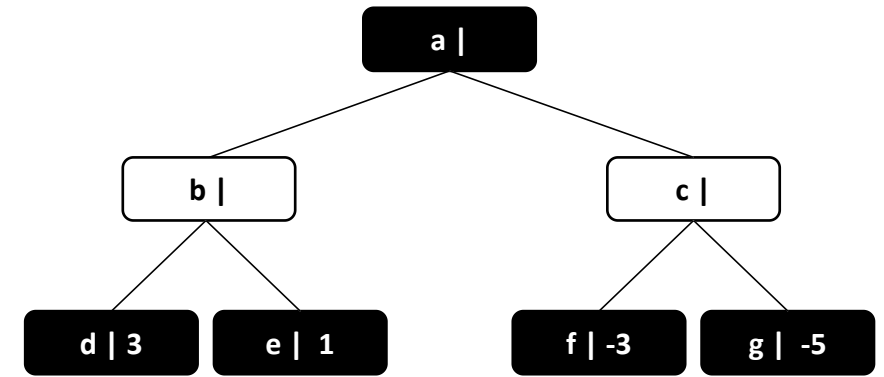
b: 1

isLeafNode(b): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(-\infty, 1)$ // we can calculate and get 1

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : $-\infty$

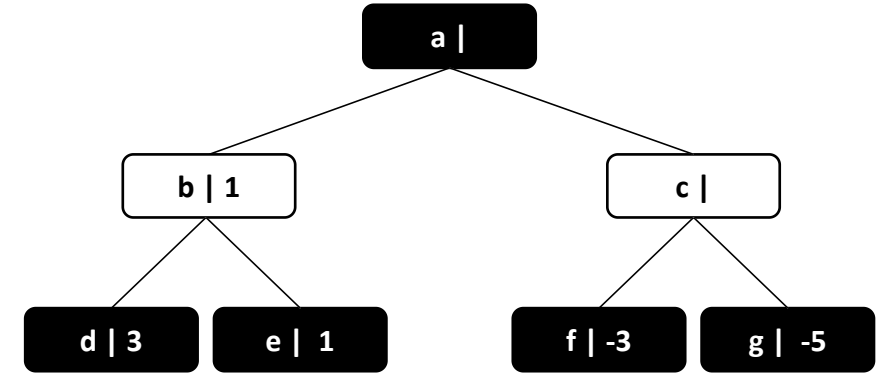
β : 1

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

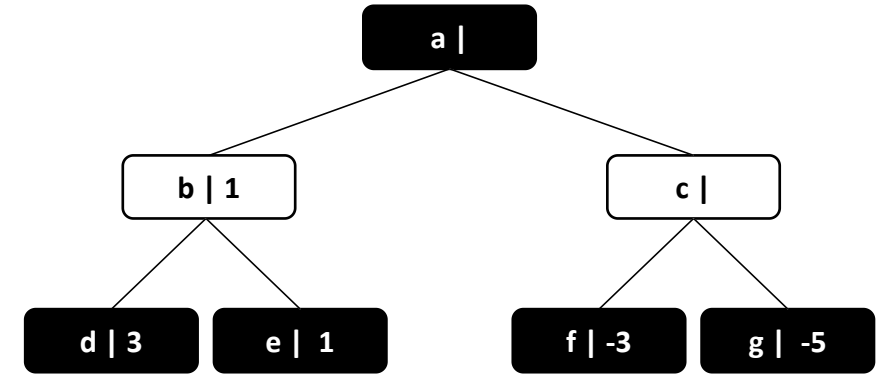
β : 3

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β // $\alpha \geq \beta$ is false

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

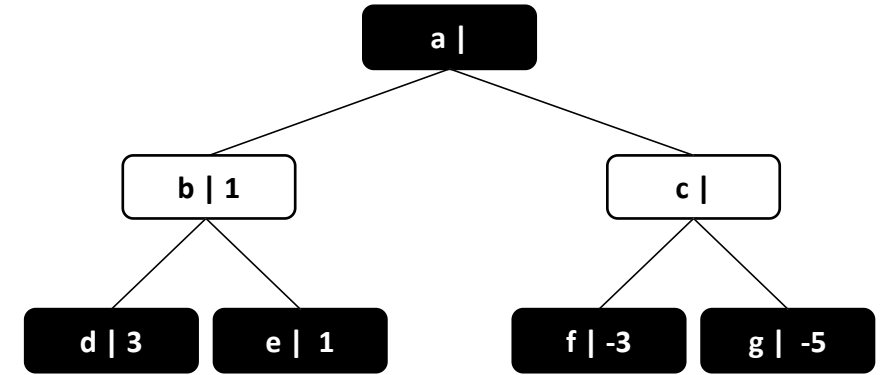
β : 3

isLeafNode(a): false

childrenNodes: b, c

first(childrenNodes): b

rest(childrenNodes): c



Start: AlphaBetaMax(**a**, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(**a**) is true THEN Return MinMaxValue(**a**)

childrenNodes = getChildren(**a**)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: **a**

α : 1

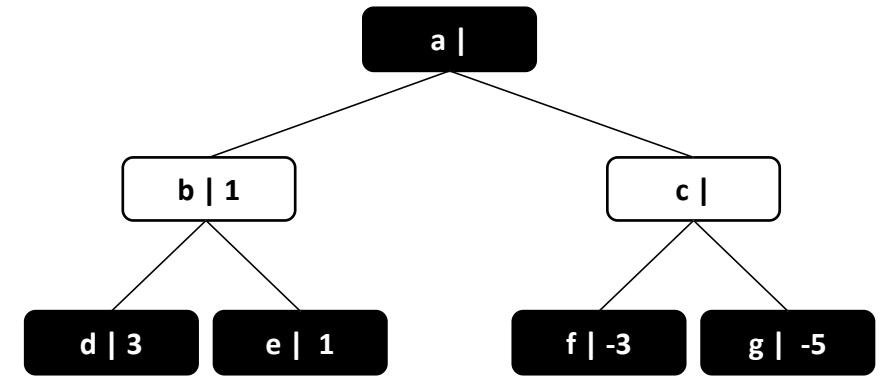
β : 3

isLeafNode(**a**): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

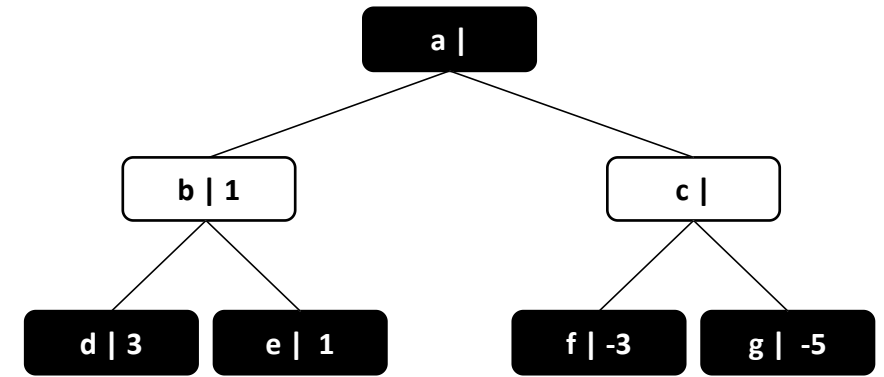
β : 3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

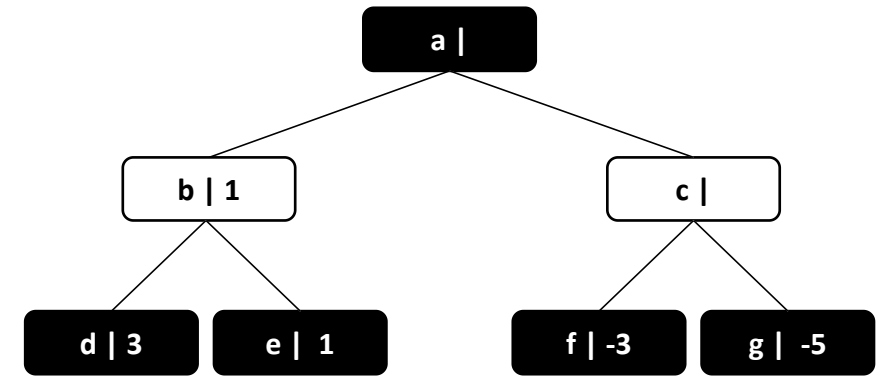
rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

```
AlphaBetaMax(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(a) is true THEN Return MinMaxValue(a)  
childrenNodes = getChildren(a)  
WHILE childrenNodes is NOT empty //not empty - go on  
     $\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$   
    IF  $\alpha \geq \beta$  THEN Return  $\beta$   
    childrenNodes = rest(childrenNodes)  
Return  $\alpha$ 
```

Node: a α : 1 b: 3
isLeafNode(a): false
childrenNodes: c
first(childrenNodes): c
rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3)$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

Node: a

α : 1

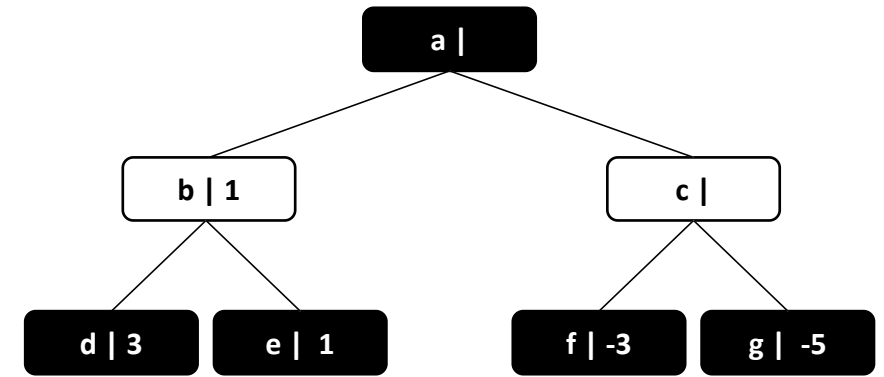
β : 3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: a

α : 1

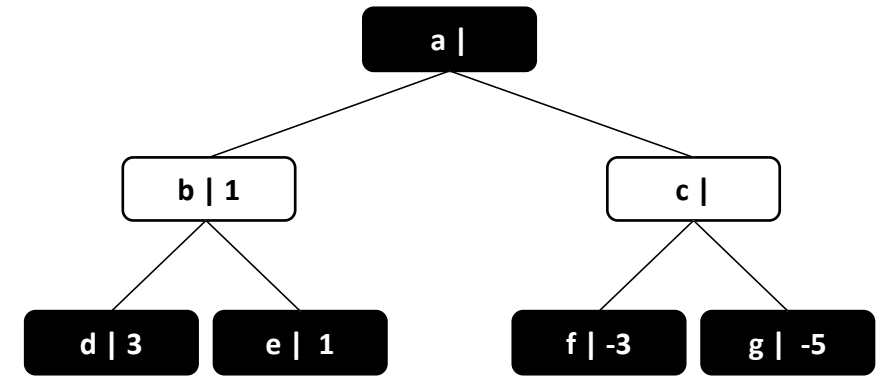
β : 3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

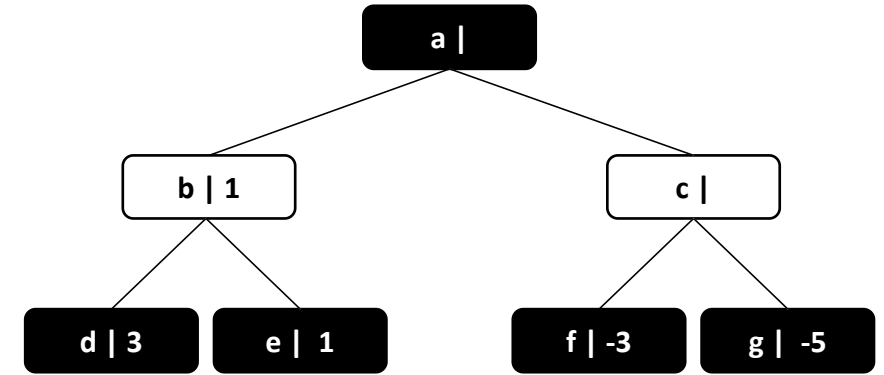
childrenNodes = rest(childrenNodes)

Return β

Node: c

α : 1

b: 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c) //false

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

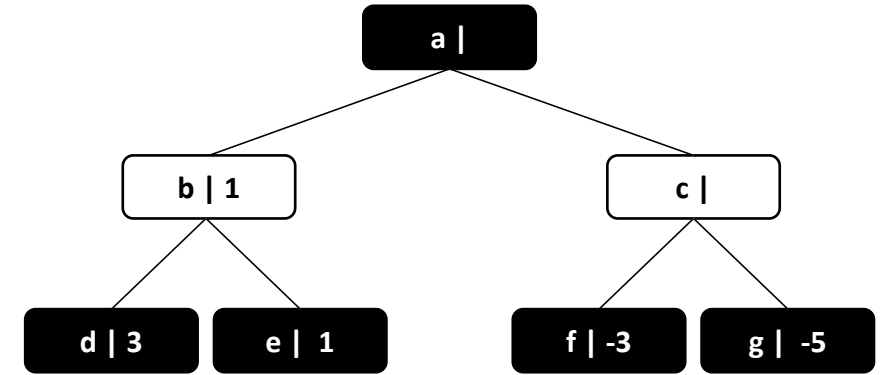
Return β

Node: c

α : 1

β : 3

isLeafNode(c): false



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: c

α : 1

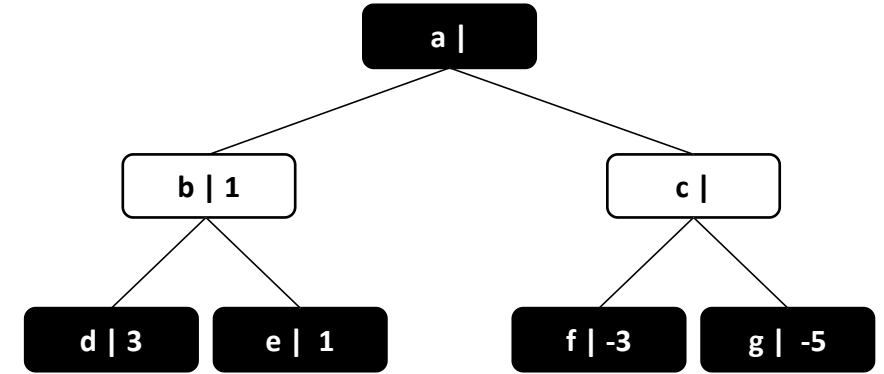
β : 3

isLeafNode(c): false

childrenNodes: f, g

first(childrenNodes): f

rest(childrenNodes): g



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: c

α : 1

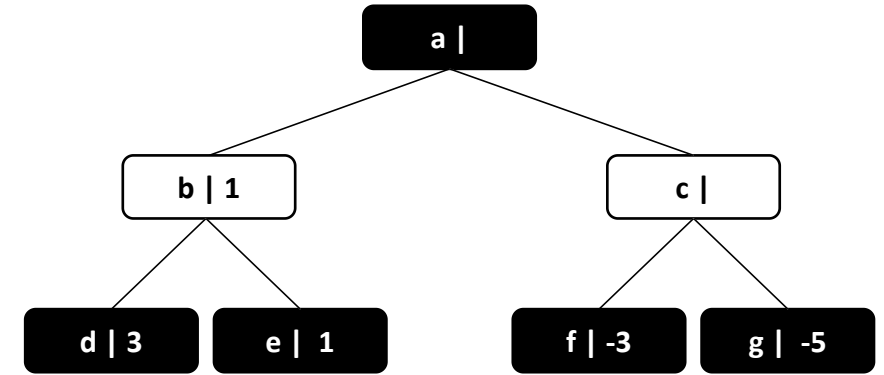
β : 3

isLeafNode(c): false

childrenNodes: f, g

first(childrenNodes): f

rest(childrenNodes): g



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(\beta, \text{AlphaBetaMax}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

Node: c

α : 1

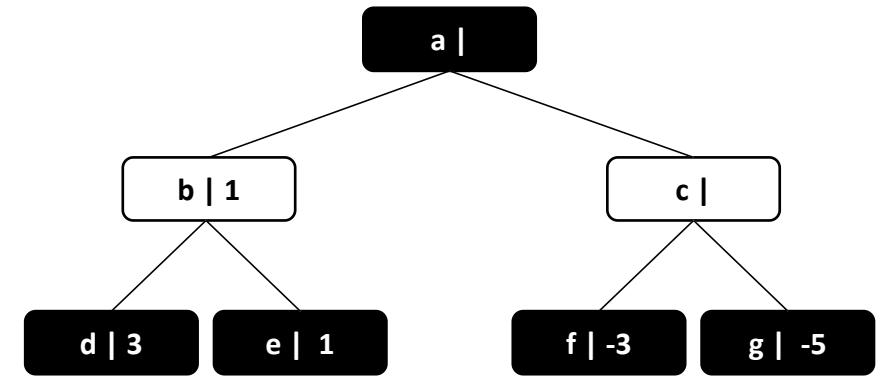
β : 3

isLeafNode(c): false

childrenNodes: f, g

first(childrenNodes): f

rest(childrenNodes): g



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3)$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(f, 1, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

Node: c

α : 1

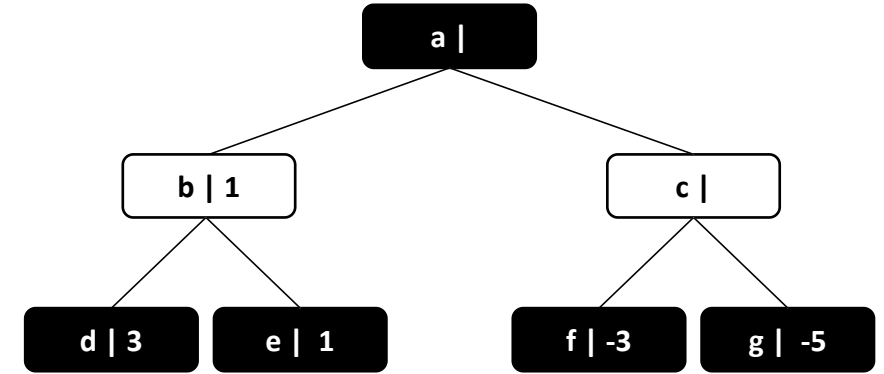
β : 3

isLeafNode(c): false

childrenNodes: f, g

first(childrenNodes): f

rest(childrenNodes): g



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(f, 1, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(Node) is true THEN Return MinMaxValue(Node)

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: c

α : 1

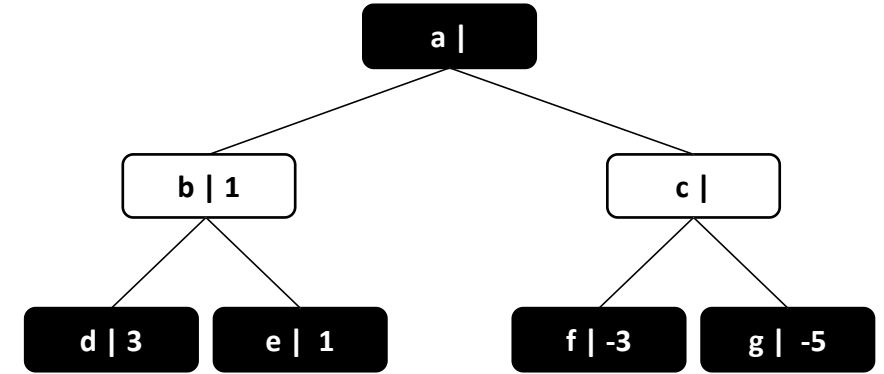
β : 3

isLeafNode(c): false

childrenNodes: f, g

first(childrenNodes): f

rest(childrenNodes): g



Start: AlphaBetaMax(a, $-\infty$, ∞)

```
AlphaBetaMax(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(a) is true THEN Return MinMaxValue(a)  
childrenNodes = getChildren(a)  
WHILE childrenNodes is NOT empty //not empty - go on  
     $\alpha = \max(1, \text{AlphaBetaMin}(c, 1, 3))$   
    IF  $\alpha \geq \beta$  THEN Return  $\beta$   
    childrenNodes = rest(childrenNodes)  
Return  $\alpha$ 
```

We can't calculate that yet.
We have to go deeper first.

```
AlphaBetaMin(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(c) is true THEN Return MinMaxValue(c)  
childrenNodes = getChildren(c)  
WHILE childrenNodes is NOT empty //not empty - go on  
     $\beta = \min(3, \text{AlphaBetaMax}(f, 1, 3))$   
    IF  $\beta \leq \alpha$  THEN Return  $\alpha$   
    childrenNodes = rest(childrenNodes)  
Return  $\beta$ 
```

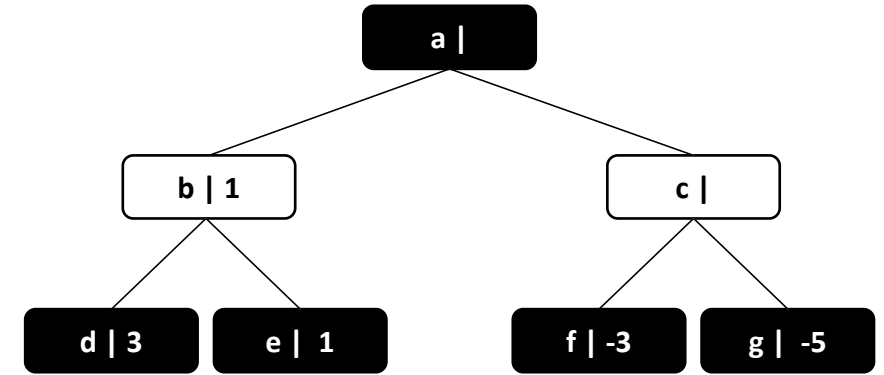
We can't calculate that yet.
We have to go deeper first.

```
AlphaBetaMax(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(f) is true THEN Return MinMaxValue(f) //true!  
childrenNodes = getChildren(Node)  
WHILE childrenNodes is NOT empty  
     $\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$   
    IF  $\alpha \geq \beta$  THEN Return  $\beta$   
    childrenNodes = rest(childrenNodes)  
Return  $\alpha$ 
```

Node: f
isLeafNode(f): true

α : 1

β : 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

```
AlphaBetaMax(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(a) is true THEN Return MinMaxValue(a)  
childrenNodes = getChildren(a)  
WHILE childrenNodes is NOT empty //not empty - go on  
     $\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$   
    IF  $\alpha \geq \beta$  THEN Return  $\beta$   
    childrenNodes = rest(childrenNodes)  
Return  $\alpha$ 
```

We can't calculate that yet.
We have to go deeper first.

```
AlphaBetaMin(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(c) is true THEN Return MinMaxValue(c)  
childrenNodes = getChildren(c)  
WHILE childrenNodes is NOT empty //not empty - go on  
     $\beta = \min(3, \text{AlphaBetaMax}(f, 1, 3))$   
    IF  $\beta \leq \alpha$  THEN Return  $\alpha$   
    childrenNodes = rest(childrenNodes)  
Return  $\beta$ 
```

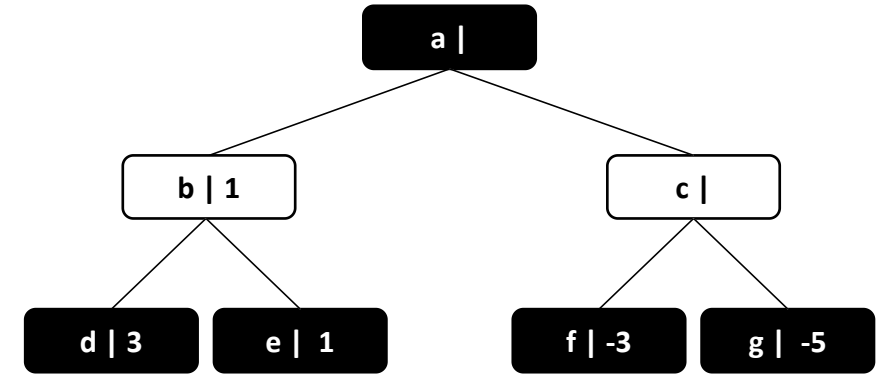
We can't calculate that yet.
We have to go deeper first.

```
AlphaBetaMax(Node,  $\alpha$ ,  $\beta$ )  
IF isLeafNode(f) is true THEN Return -3 //true!  
childrenNodes = getChildren(Node)  
WHILE childrenNodes is NOT empty  
     $\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$   
    IF  $\alpha \geq \beta$  THEN Return  $\beta$   
    childrenNodes = rest(childrenNodes)  
Return  $\alpha$ 
```

Node: f
isLeafNode(f): true

α : 1

β : 3



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, \text{AlphaBetaMax}(f, 1, 3))$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

Return β

-3
returned

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMax(Node, α , β)

IF isLeafNode(f) is true THEN Return -3 //true!

childrenNodes = getChildren(Node)

WHILE childrenNodes is NOT empty

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

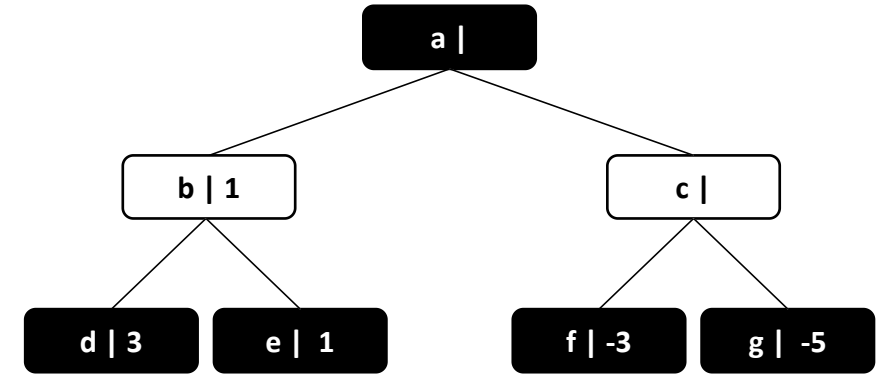
Return α

Node: f

α : 1

β : 3

isLeafNode(f): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3)$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = \min(3, -3)$ // we can calculate and get -3

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

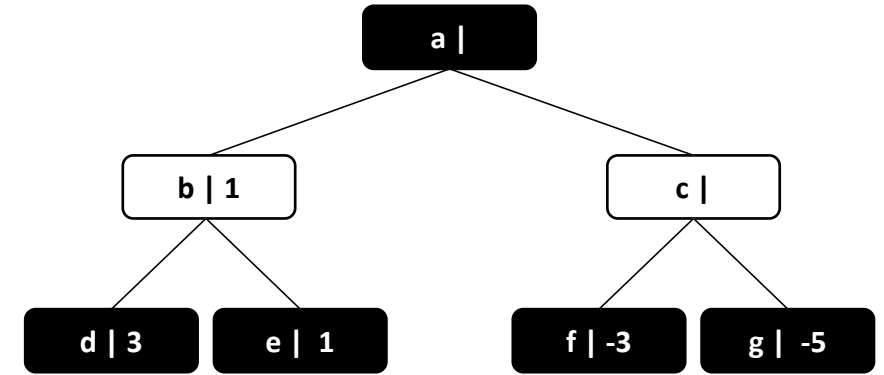
Return β

Node: f

α : 1

β : 3

isLeafNode(f): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3)$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = -3$

IF $\beta \leq \alpha$ THEN Return α

childrenNodes = rest(childrenNodes)

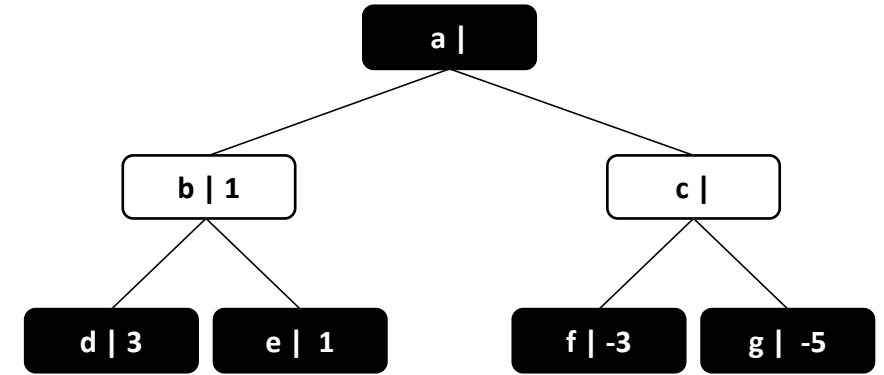
Return β

Node: f

α : 1

β : -3

isLeafNode(f): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = -3$

IF $\beta \leq \alpha$ THEN Return α //true! return alpha

childrenNodes = rest(childrenNodes)

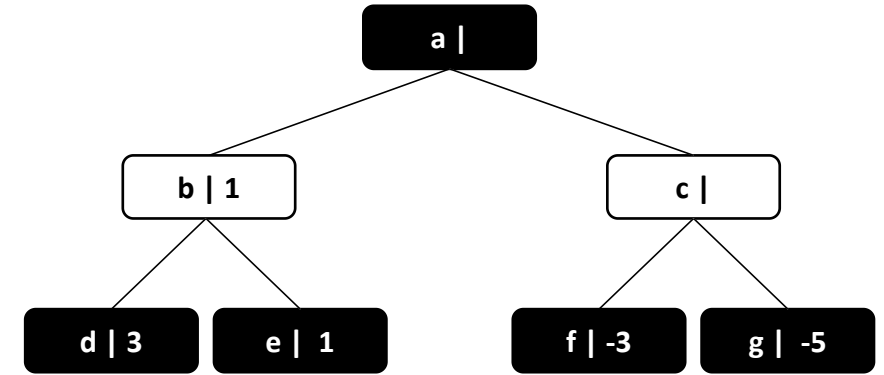
Return β

Node: f

α : 1

β : -3

isLeafNode(f): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, \text{AlphaBetaMin}(c), 1, 3))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

1

returned

We can't calculate that yet.
We have to go deeper first.

AlphaBetaMin(Node, α , β)

IF isLeafNode(c) is true THEN Return MinMaxValue(c)

childrenNodes = getChildren(c)

WHILE childrenNodes is NOT empty //not empty - go on

$\beta = -3$

IF $\beta \leq \alpha$ THEN Return α //true! prune / return alpha

childrenNodes = rest(childrenNodes)

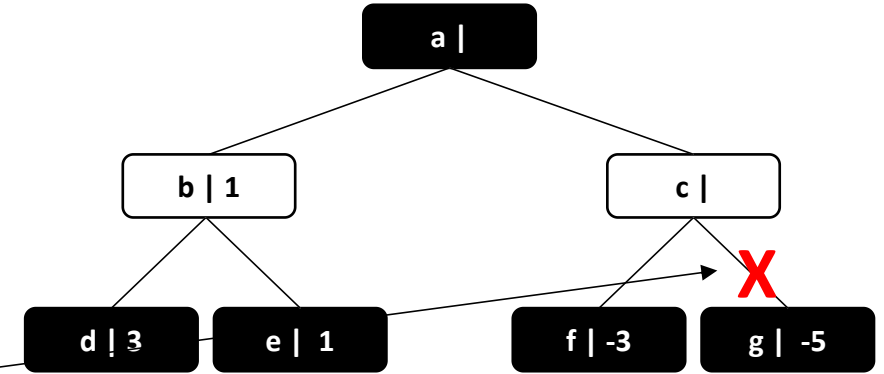
Return β

Node: f

α : 1

β : -3

isLeafNode(f): true



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = \max(1, 1)$ // we can calculate and get 1

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

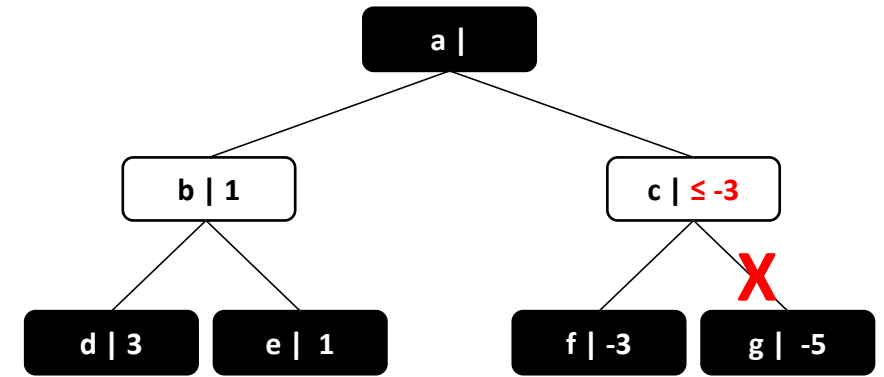
β : -3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

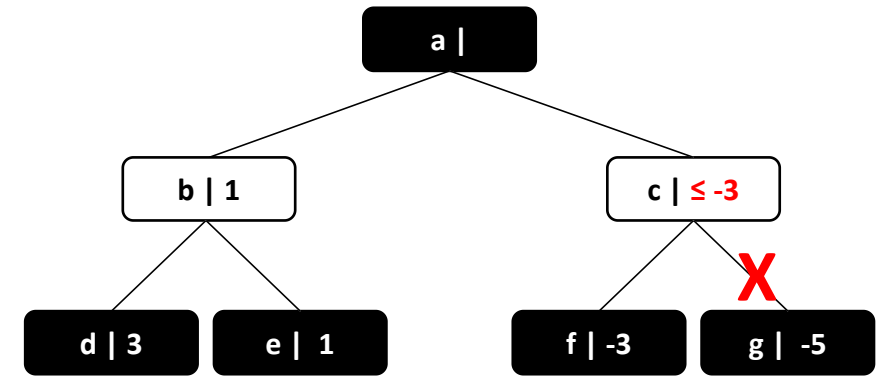
β : -3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β // false

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

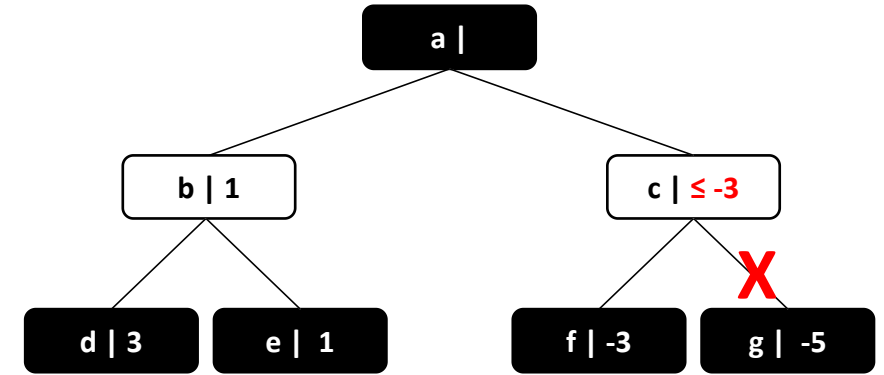
β : -3

isLeafNode(a): false

childrenNodes: c

first(childrenNodes): c

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty //not empty - go on

$\alpha = 1$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

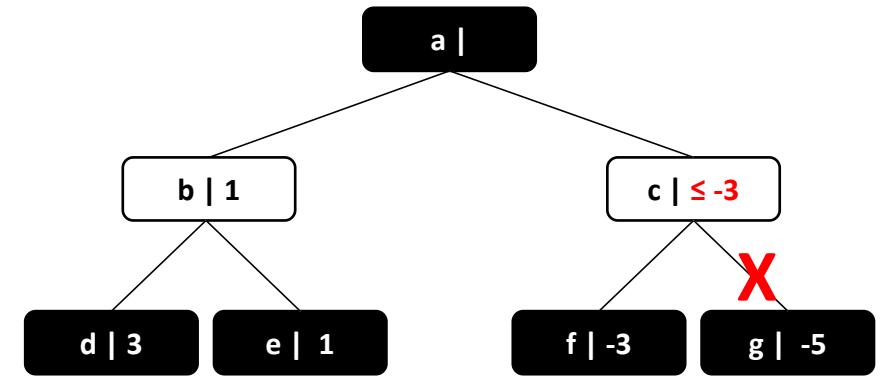
β : -3

isLeafNode(a): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty // empty! leave the loop!

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

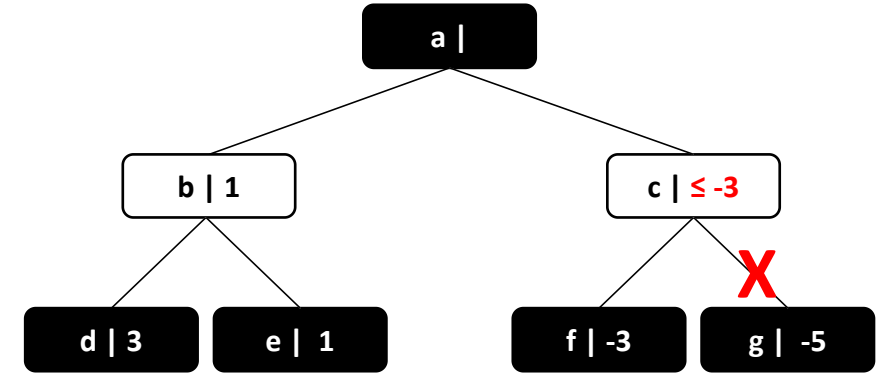
β : -3

isLeafNode(a): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty // empty! leave the loop!

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

Node: a

α : 1

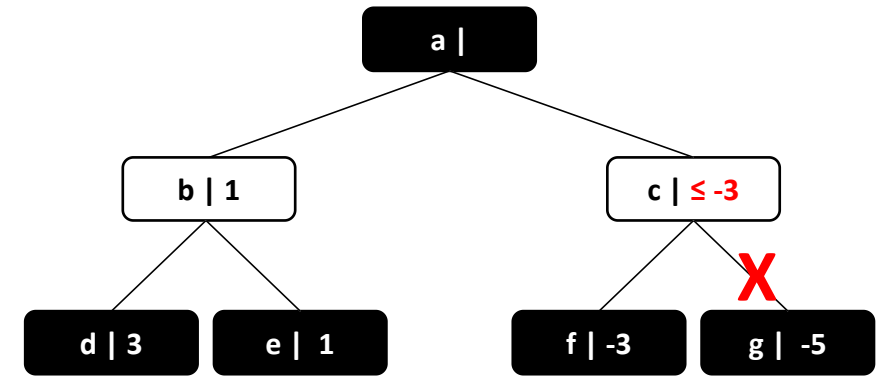
β : -3

isLeafNode(a): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty // empty! leave the loop!

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

1

returned

Node: a

α : 1

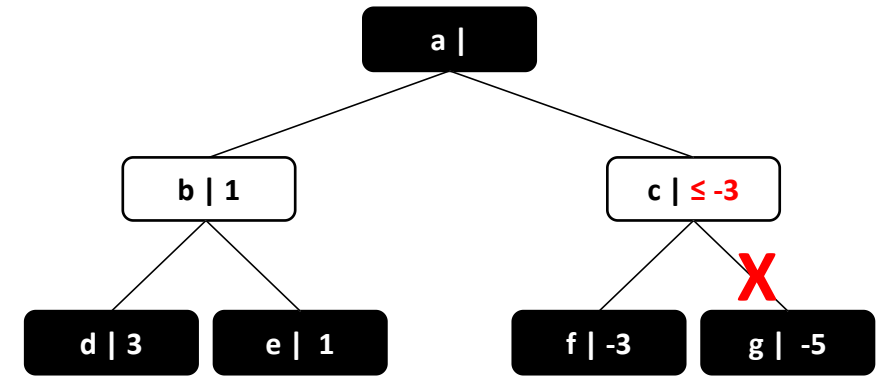
b: -3

isLeafNode(a): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start: AlphaBetaMax(a, $-\infty$, ∞)

AlphaBetaMax(Node, α , β)

IF isLeafNode(a) is true THEN Return MinMaxValue(a)

childrenNodes = getChildren(a)

WHILE childrenNodes is NOT empty // empty! leave the loop!

$\alpha = \max(\alpha, \text{AlphaBetaMin}(\text{first}(\text{childrenNodes}), \alpha, \beta))$

IF $\alpha \geq \beta$ THEN Return β

childrenNodes = rest(childrenNodes)

Return α

1

returned

Node: a

α : 1

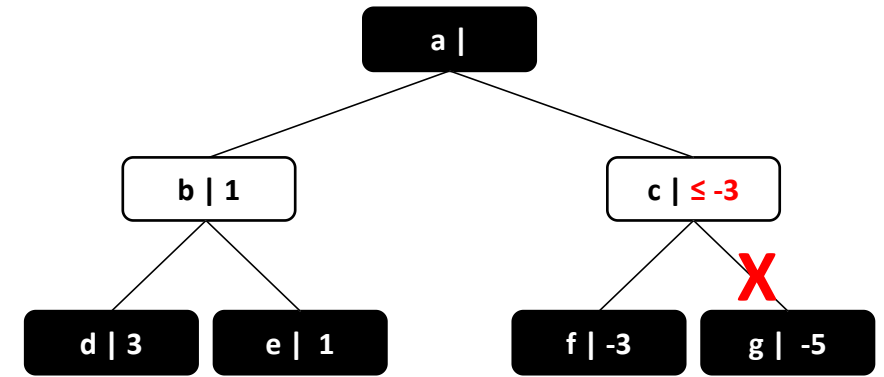
b: -3

isLeafNode(a): false

childrenNodes: empty

first(childrenNodes): none

rest(childrenNodes): empty



Start:AlphaBetaMax(a, -∞, ∞)

Node: a

α : 1

β : -3

