1 Preliminaries

1.1 Color index

abbreviation
sublevel
partial result

1.2 Abbreviation

BR	break
CA	case
CO	construct
CT	continue
FA	false
FO	for
IF	if
IN	infinity
NU	null
SW	switch
TR	true
WH	while

1.3 Notation

1.3.1 Object

1. **Empty object**

$\emptyset$	null
	[ ]
	“”

2. **Interior**

O°	interior of O

1.3.2 Operator

1. **Operation**

$\boxed{\circ}\begin{bmatrix} a \\ b \end{bmatrix}$	$a \circ b$

2. **Assignment**

$a \overset{\circ}{=} b$	$a = a \circ b$

	$a \overset{-}{=} b$	$a = a - b$
	$a \overset{+}{=} b$	$\boxed{\vee}\begin{bmatrix} a = a + b \\ \text{add } b \text{ to an array or set } a \end{bmatrix}$
$\boxed{1}$	$a \overset{\times}{=} b$	$a = a \times b$
	$a \overset{\div}{=} b$	$a = a \div b$
	$a \overset{\cup}{=} b$	$a = a \cup b$
	$a \overset{\cap}{=} b$	$a = a \cap b$

3. **Increment**

$\overset{+}{+a}$	$++a$
$a\overset{+}{+}$	$a++$
$\overset{-}{-a}$	$--a$
$a\overset{-}{-}$	$a--$

4. **Comparison**

$a \equiv b$	a is equal to b
$b \neq b$	a is not equal to b

5. **Multiplication**

$a \times matrix$	scalar product
$coordinates \times coordinates$	inner product
$matrix \times matrix$	matrix multiplication

6. **Negation**

$\neg a$	not a

7. **Absolute value** | |

$\lvert real \rvert$	absolute value of $real$
$\lvert array \rvert$	number of elements in $array$
$\lvert matrix \rvert$	determinant of $matrix$
$\lvert \mathbf{xy} \rvert$	norm of **xy**
$\lvert \mathbf{Point2D} \rvert$	norm of **Point2D**
$\lvert \mathbf{Segment2D} \rvert$	length of **Segment2D**

1.3.3 Array

1. $\left[\begin{array}{c|c} value\_1 & value\_2 \end{array}\right]$
2. $\left[\begin{array}{c|c} key\_1 & key\_2 \\ \hline value\_1 & value\_2 \end{array}\right]$
3. $\left[\begin{array}{c|c} array \begin{bmatrix} key\_1 \\ key\_2 \end{bmatrix} & array[key\_1][key\_2] \end{array}\right]$
4. We regard array as a set if there is no confusion.

1.3.4 Control

1. **Conditional**

IF	condition 1	excution 1
	condition 2	excution 2
	excution 3	

2. **For**

1

FO	condition
	excution

2

FO	*initial*	*condition*	*increment*
	excution		

3

FO	*array*	*value*
	excution	

4

FO	*array*	*key*	*value*
	excution		

3. **Switch**

SW	*value*	
	value 1	excution 1
	value 2	excution 2
	excution 3	

4. **While**

WH	condition
	excution

1.3.5 Function

1. **Define**

name	*variable*
	algorithm

2. **Call**

name $\begin{pmatrix} value \\ value \end{pmatrix}$	name$(value, value)$

1.3.6 Class

1. **Define**

name	*property*
	construct
	method

1	construct	CO	*variable*
			algorithm
	method	name	*variable*
			algorithm

2. **Assign**

1 **name** (*value*)

2 We regard **name** as **name**() if there is no confusion

3. **Access**

1	**name** ▶ *sub*	
	name ▶ *sub*_1 ▶ *sub*_2	**name** ▶ $\begin{bmatrix} sub\_1 \\ sub\_2 \end{bmatrix}$

2 We regard method as method() if there is no confusion

4. **Extend**

this ◀ *parent*

1.4 Types

1.4.1 Boolean

1. Every expression is either TH or FA.

2 JSON

2.1 Cartesian2D

menu	
elements	
axis	
grid	
font	color: text, background size decoration: underline, bold, italic
position	left, top
margin	top, bottom, left, right
size	
ratio	
rotation	
flip	x, y
bounds	x: min, max y: min, max
display	
elmAlign	
interaction	zooming, panning, rotation

2.2 Angle2D

coords	center: x, y start: x, y end: x, y
style	color height curve rigntAngle fill dash arrow: start, end marker markerHegiht
interaction	interactive, selected

2.3 Arc2D

measure	
coords	center: x, y start: x, y
angle	
style	color dash arrow: start, end marker1 marker2 marker3
interaction	

2.4 Curve2D

domain	x: min, max y: min, max
points	[x, y]
equation	
style	color dash
interaction	movable-mode selectable removable domain

2.5 Face2D

coords	[x, y]
interaction	interactive selected selectable movable
style	color

2.6 Label2D

label	
coords	x, y
target	x, y
arrowColor	
dash	

2.7 Measure2D

style	height color dash arrow: x, y marker

2.8 MeasureArc2D

style	height	
	color	
	dash	
	arrow	start end
	marker	
	handle	
	handleDash	
interaction	selectable	
	movable	

2.9 Point2D

coord	x	
	y	
style	fill	
	color	
interaction	selected	
	selectable	
	movable	
	removable	

2.10 Region2D

curves	[object]
fill	[< fill | > fill]
style	color
interaction	selectable

2.11 Segment2D

measure		
coords	start	x y
	end	x y
style	color	
	dash	
	arrow	start end
	marker1	
	marker2	
	marker3	
interaction	interactive	
	selected	

3 Structure

3.1 Cartesian2D

type	Cartesian2D
elements	[ ]
rotation	
size	
zoom	

3.2 Angle-free

type	arc-free		
_id			
struct	coords	center	x y
		start	x y
		end	x y
	style	height	
		rightAngle	

3.3 Arc-free

type	angle-free		
_id			
struct	coords	center	x y
		start	x y
	angle		
	height		

3.4 Curve-free

type	curve-free
_id	
struct	

3.5 Face-free

type	face-free		
_id			
struct	coords	[	x y

3.6 Label-free

type	label-free		
_id			
struct	coords	x y	
	target	x y	
	label	type content	
	dependency	[ ]	

3.7 Point-free

type	point-free		
_id			
struct	coord	x y	

3.8 Region-free

type	region-free
_id	
struct	

3.9 Segment-free

type	segment-free			
_id				
struct	coords	start	x	
			y	
		end	x	
			y	
	measure	height		

4 Basic

4.1 Math

4.1.1 numeric

1. **Definition**

numeric	
	method

2. **Note**

   We omit **numeric** ▶ if there is no confusioion.

3. **Method**

1	mod	compute remainder of division *real_1* by *real_2*
	mod	real_1 | real_2
		RE | $-\left[real\_1 \;\middle|\; \times \left[|real\_2| \;\middle|\; \div \left[real\_1 \;\middle|\; |real\_2| \right] \right] \right]$

2	sign	sign of *real*
	sign	*real* | $sign\_zero = 0$
		IF | $real < 0$ | RE | -1
		$real \equiv 0$ | RE | $sign\_zero$
		RE | 1

3	vector_angle	vector angle of point $\left[x \;\middle|\; y \right]$
	vector_angle	x | y
		RE | $\mathrm{atan2}\left(\begin{matrix} y \\ x \end{matrix} \right)$

4.1.2 angle (radian)

1. **Definition**

angle	
	method

2. **Note**

   We omit **angle** ▶ if there is no confusioion.

3. **Method**

1	polar	polar coordinate angle of *angle*
	polar	*angle* | $angle\_start = 0$
		RE | $+\left[angle\_start \;\middle|\; \mathrm{mod}\left(\begin{matrix} -\left[\begin{matrix} angle \\ angle\_start \end{matrix} \right] \\ 2\pi \end{matrix} \right) \right]$

4.1.3 xy

1. **Descritption**

   [1] A class to reprensent coordinates in xy-plane.

   [2] For simplicity, we regard it as a point $\left[\begin{array}{c|c} x & y \end{array}\right]$ in xy-plane.

2. **Definition**

xy	$type = \mathsf{xy}$
	x
	y
	method

3. **Method**

[1] construct

CO: a | b | $type$

SW: $type$

Cartesian	$this = \left[\begin{array}{c\|c} a & b \end{array}\right]$
polar	$this = \left[\begin{array}{c\|c} a\cos(b) & a\sin(b) \end{array}\right]$

[2] angle | vector angle

angle

RE: $\text{vector\_angle}\begin{pmatrix} this \blacktriangleright x \\ this \blacktriangleright y \end{pmatrix}$

[3] rotation | rotate arround *center* with angle *angle*

rotation: *center* | *angle*

RE: $+\left[\begin{array}{l} center \\ \times\left[\begin{array}{l} -\left[\begin{array}{l} this \\ center \end{array}\right] \\ \begin{bmatrix} \cos(angle) & -\sin(angle) \\ \sin(angle) & \cos(angle) \end{bmatrix} \end{array}\right] \end{array}\right]$

[4] vector_angle | vetctor angle of $xy - this$

vector_angle: xy

RE: $\text{vector\_angle}\left(\begin{array}{l} -\left[\begin{array}{l} xy \blacktriangleright y \\ this \blacktriangleright y \end{array}\right] \\ -\left[\begin{array}{l} xy \blacktriangleright x \\ this \blacktriangleright x \end{array}\right] \end{array}\right)$

4.1.4 Bezier

1. **Definition**

Bezier	$start$
	mid
	end
	method

2. **Method**

[1] construct

CO: $start$ | mid | end

$=\left[\begin{array}{l} this \blacktriangleright start \\ start \end{array}\right]$

$=\left[\begin{array}{l} this \blacktriangleright mid \\ mid \end{array}\right]$

$=\left[\begin{array}{l} this \blacktriangleright end \\ end \end{array}\right]$

[2] center | center

center

$$mid\_segment = \boxed{+} \begin{bmatrix} this \blacktriangleright start \\ this \blacktriangleright end \end{bmatrix} \times \frac{1}{2}$$

$$radius = \frac{1}{2} \times \left| \boxed{-} \begin{bmatrix} mid\_segment \\ this \blacktriangleright mid \end{bmatrix} \right|$$

$$\boxed{=} \begin{bmatrix} angle \\ mid\_segment \blacktriangleright \text{vector\_angle}(this \blacktriangleright mid) \end{bmatrix}$$

$$translation = \mathbf{xy} \begin{pmatrix} radius \\ angle \\ \text{polar} \end{pmatrix}$$

RE $\quad \boxed{+} \begin{bmatrix} mid\_segment \\ translation \end{bmatrix}$

3 | coordinates | coordinates at t

coordinates | t

RE $\quad \boxed{+} \begin{bmatrix} \boxed{\times} \begin{bmatrix} (1-t)^2 \\ this \blacktriangleright start \end{bmatrix} \\ \boxed{\times} \begin{bmatrix} 2(1-t)t \\ this \blacktriangleright mid \end{bmatrix} \\ \boxed{\times} \begin{bmatrix} t^2 \\ this \blacktriangleright end \end{bmatrix} \end{bmatrix}$

4 | height | height

height

RE $\quad \left| \boxed{+} \begin{bmatrix} this \blacktriangleright start \\ this \blacktriangleright end \\ -2 \times this \blacktriangleright mid \end{bmatrix} \right| \times \frac{1}{4}$

5 | interval | **Interval** with given $axis$

interval | $axis$

RE $\quad this \blacktriangleright \begin{bmatrix} \text{quadratic}(axis) \\ \text{image} \left(\mathbf{Interval} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) \end{bmatrix}$

6 | quadratic | quadratic

quadratic | $axis$

$$a = \boxed{+} \begin{bmatrix} this \blacktriangleright \begin{bmatrix} start \\ axis \end{bmatrix} \\ -2 \times this \blacktriangleright \begin{bmatrix} mid \\ axis \end{bmatrix} \\ this \blacktriangleright \begin{bmatrix} end \\ axis \end{bmatrix} \end{bmatrix}$$

$$b = \boxed{-} \begin{bmatrix} this \blacktriangleright \begin{bmatrix} mid \\ axis \end{bmatrix} \\ this \blacktriangleright \begin{bmatrix} end \\ axis \end{bmatrix} \end{bmatrix} \times 2$$

$$c = this \blacktriangleright \begin{bmatrix} start \\ axis \end{bmatrix}$$

RE $\quad \mathbf{Quadratic} \begin{pmatrix} a \\ b \\ c \end{pmatrix}$

7 | vector_tangent | tangent vector at t

vector_tangent	t
	$= \begin{bmatrix} quadratic\_x \\ this \blacktriangleright \text{quadratic(x)} \end{bmatrix}$
	$= \begin{bmatrix} quadratic\_y \\ this \blacktriangleright \text{quadratic(y)} \end{bmatrix}$
	$x = + \begin{bmatrix} 2t \times quadratic\_x \blacktriangleright a \\ quadratic\_x \blacktriangleright b \end{bmatrix}$
	$y = + \begin{bmatrix} 2t \times quadratic\_y \blacktriangleright a \\ quadratic\_y \blacktriangleright b \end{bmatrix}$
	RE | $\mathbf{xy} \begin{pmatrix} x \\ y \end{pmatrix}$

4.1.5 index

1. **Definition**

index	$type =$ index
	CO | $value$
	IF | is_array($value$) | FO | $index \in value$
	$= \begin{bmatrix} this \blacktriangleright index \\ \emptyset \end{bmatrix}$
	is_string($value$) | SW | value
	direction | $= \begin{bmatrix} this \blacktriangleright left \\ \emptyset \end{bmatrix}$
	$= \begin{bmatrix} this \blacktriangleright right \\ \emptyset \end{bmatrix}$
	position | $= \begin{bmatrix} this \blacktriangleright start \\ \emptyset \end{bmatrix}$
	$= \begin{bmatrix} this \blacktriangleright end \\ \emptyset \end{bmatrix}$
	sign | $= \begin{bmatrix} this \blacktriangleright positive \\ \emptyset \end{bmatrix}$
	$= \begin{bmatrix} this \blacktriangleright negative \\ \emptyset \end{bmatrix}$

4.1.6 Interval

1. **Definition**

Interval	$type =$ Interval
	$min = \emptyset$
	$max = \emptyset$
	method

2. **Method**

1 | construct |

CO	min | max
IF	$min \leq max$
	$= \begin{bmatrix} this \blacktriangleright min \\ min \end{bmatrix}$
	$= \begin{bmatrix} this \blacktriangleright max \\ max \end{bmatrix}$
	$this = \emptyset$

2 | extension | minimal **Interval** that contains $this$ and $real$ |

extension	real
RE	**Interval** $\begin{pmatrix} \min \begin{pmatrix} this \blacktriangleright min \\ real \end{pmatrix} \\ \max \begin{pmatrix} this \blacktriangleright max \\ real \end{pmatrix} \end{pmatrix}$

3	indicator	determine if *this* contains *real*

indicator	real		
IF	$real \in this$	RE	1
	RE	0	

4	translation	translated with *real*

translation	*real*
$=$	$\begin{bmatrix} this\_translated \\ this \end{bmatrix}$
$\pm$	$\begin{bmatrix} this\_translated \blacktriangleright min \\ real \end{bmatrix}$
$\pm$	$\begin{bmatrix} this\_translated \blacktriangleright max \\ real \end{bmatrix}$
RE	*this_translated*

4.1.7 Interval_angle

1. **Definition**

Interval_angle	$type =$ Interval_angle
	$start = \emptyset$
	$size = \emptyset$
	method

2. **Method**

1	construct	

CO	*start*	*size*
$this \blacktriangleright start = +$	$\begin{bmatrix} start \\ \times \begin{bmatrix} \min \begin{pmatrix} 0 \\ \text{sign}(size) \end{pmatrix} \\ size \end{bmatrix} \end{bmatrix}$	
$=$	$\begin{bmatrix} this \blacktriangleright size \\ \lvert size \rvert \end{bmatrix}$	

2	end	end

end			
IF	$this \equiv \emptyset$	RE	NU
RE	$+ \begin{bmatrix} this \blacktriangleright start \\ this \blacktriangleright size \end{bmatrix}$		

3	mid	mid angle

mid	
RE	$+ \begin{bmatrix} this \blacktriangleright start \\ \frac{1}{2} \times this \blacktriangleright size \end{bmatrix}$

4.1.8 Matrix

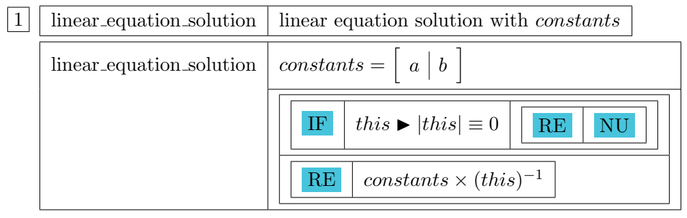
1. **Description**

We regard a 2×2 matrix M as a 2-dimensional matrix with $M = \begin{bmatrix} M[0][0] & M[0][1] \\ M[1][0] & M[1][1] \end{bmatrix}$

2. **Definition**

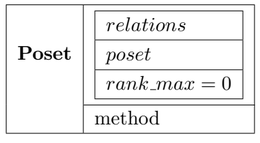
Matrix ◀ Array	*array*
	method

3. **Method**

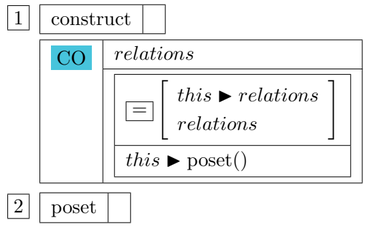


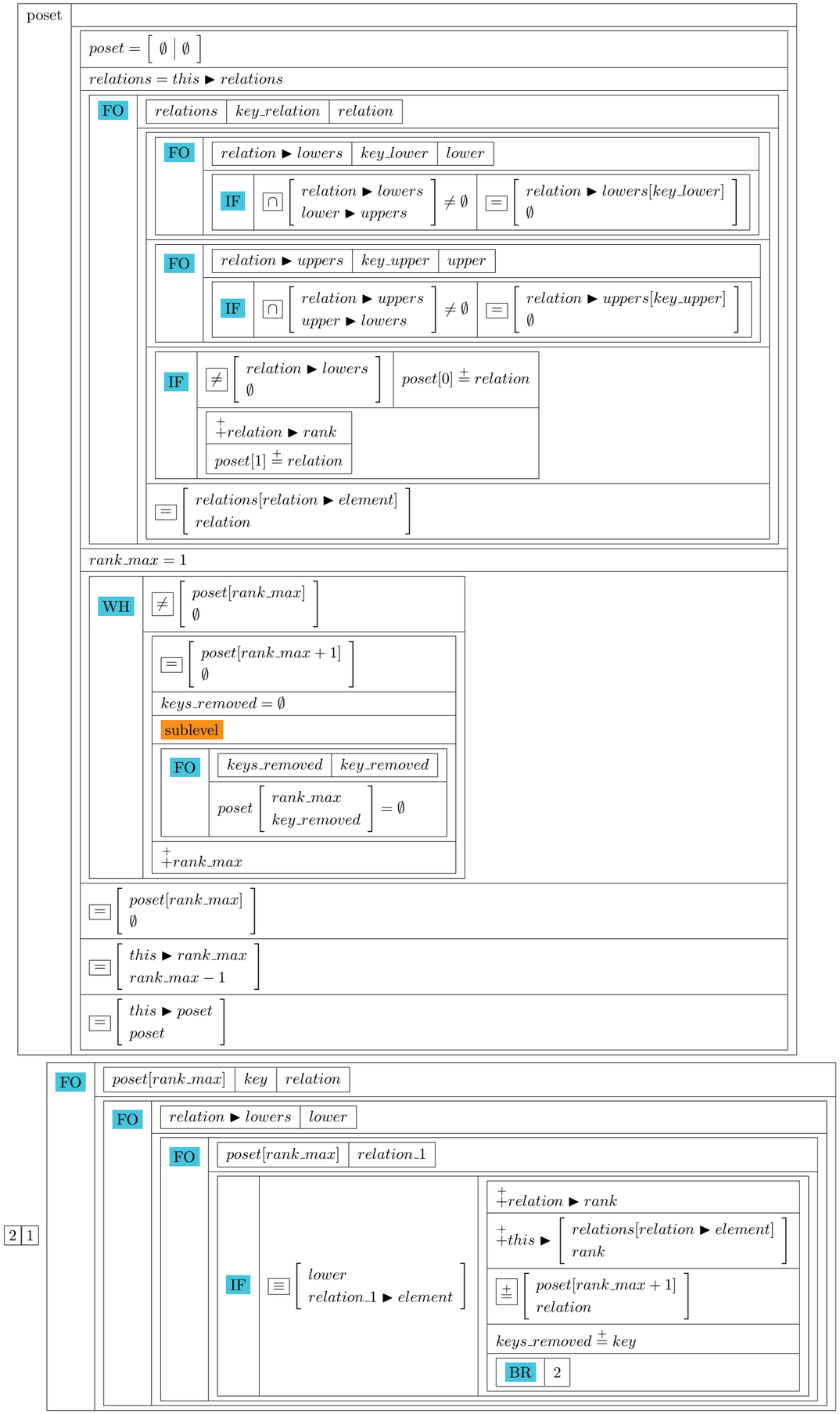
4.1.9 Poset

1. **Definition**



2. **Method**





4.1.10 Relation

1. **Definiton**

Relation	*element*
	lowers
	uppers
	rank
	method

2. **Method**

1	construct	

CO	*element*	$uppers = \emptyset$	$lowers = \emptyset$	$rank = 0$

$$= \begin{bmatrix} this \blacktriangleright element \\ element \end{bmatrix}$$

$$= \begin{bmatrix} this \blacktriangleright uppers \\ uppers \end{bmatrix}$$

$$= \begin{bmatrix} this \blacktriangleright lowers \\ lowers \end{bmatrix}$$

$$= \begin{bmatrix} this \blacktriangleright rank \\ rank \end{bmatrix}$$

4.1.11 Quadratic

1. **Definition**

Quadratic	$type =$ Quadratic
	a
	b
	c
	method

2. **Method**

1	construct	

CO	a	b	c

$$= \begin{bmatrix} this \blacktriangleright a \\ a \end{bmatrix}$$

$$= \begin{bmatrix} this \blacktriangleright b \\ b \end{bmatrix}$$

$$= \begin{bmatrix} this \blacktriangleright c \\ c \end{bmatrix}$$

2	discriminat	discriminant

discriminant	

RE

$$- \begin{bmatrix} (this \blacktriangleright a)^2 \\ \times \begin{bmatrix} 4 \times this \blacktriangleright a \\ this \blacktriangleright c \end{bmatrix} \end{bmatrix}$$

3	evaluation	eveluate with x

evaluation	x

RE

$$+ \begin{bmatrix} x^2 \times this \blacktriangleright a \\ x \times this \blacktriangleright b \\ this \blacktriangleright c \end{bmatrix}$$

4	image	image with *domain*

image	*domain*
$\boxed{=}\begin{bmatrix} evaluation\_min \\ this \blacktriangleright \text{evaluation}(domain \blacktriangleright min) \end{bmatrix}$	
$\boxed{=}\begin{bmatrix} evaluation\_max \\ this \blacktriangleright \text{evaluation}(domain \blacktriangleright max) \end{bmatrix}$	
$image = \textbf{Interval}()$	
$image \blacktriangleright min = \min\begin{pmatrix} evaluation\_min \\ evaluation\_max \end{pmatrix}$	
$image \blacktriangleright max = \max\begin{pmatrix} evaluation\_min \\ evaluation\_max \end{pmatrix}$	
IF $\boxed{\equiv}\begin{bmatrix} this \blacktriangleright a \\ 0 \end{bmatrix}$	RE *image*
$x\_apex = \boxed{\div}\begin{bmatrix} -2 \times this \blacktriangleright b \\ this \blacktriangleright a \end{bmatrix}$	
IF $x\_apex \notin domain$	RE *image*
$\boxed{=}\begin{bmatrix} y\_apex \\ this \blacktriangleright \text{evaluation}(x\_apex) \end{bmatrix}$	
$image \blacktriangleright min = \min\begin{pmatrix} image \blacktriangleright min \\ y\_apex \end{pmatrix}$	
$image \blacktriangleright max = \max\begin{pmatrix} image \blacktriangleright max \\ y\_apex \end{pmatrix}$	
RE *image*	

5 | preimage | preimage with *codomain*

preimage	*codomain*
$\boxed{=}\begin{bmatrix} solutions\_min \\ this \blacktriangleright \text{solution}(codomain \blacktriangleright min) \end{bmatrix}$	
$\boxed{=}\begin{bmatrix} solutions\_max \\ this \blacktriangleright \text{solution}(codomain \blacktriangleright max) \end{bmatrix}$	
$preimage = \textbf{Interval}()$	
IF $\boxed{\wedge}\begin{bmatrix} solutions\_min \equiv \emptyset \\ solutions\_max \equiv \emptyset \end{bmatrix}$	RE *preimage*
$solutions\_min \overset{+}{=}$ IN	
$solutions\_max \overset{+}{=} -$ IN	
IF $this \blacktriangleright a \equiv 0$	RE *preimage*
$y\_apex = \boxed{\div}\begin{bmatrix} \boxed{+}\begin{bmatrix} -(this \blacktriangleright b)^2 \\ \boxed{\times}\begin{bmatrix} 4 \times this \blacktriangleright a \\ this \blacktriangleright c \end{bmatrix} \end{bmatrix} \\ 4 \times this \blacktriangleright a \end{bmatrix}$	
IF $y\_apex \notin codomain$	RE *preimage*
$x\_apex = \boxed{\div}\begin{bmatrix} -2 \times this \blacktriangleright b \\ this \blacktriangleright a \end{bmatrix}$	
$preimage \blacktriangleright min = \min\begin{pmatrix} preimage \blacktriangleright min \\ x\_apex \end{pmatrix}$	
$preimage \blacktriangleright max = \min\begin{pmatrix} preimage \blacktriangleright max \\ x\_apex \end{pmatrix}$	
RE *preimage*	

6 | solution | solve $this \equiv constant = 0$

solution	*constant*
	$quadratic = \textbf{Quadratic}\left(\begin{array}{l} this \blacktriangleright a \\ this \blacktriangleright b \\ \boxed{-}\left[\begin{array}{l} this \blacktriangleright c \\ constant \end{array}\right] \end{array}\right)$
	$\boxed{=}\left[\begin{array}{l} d \\ quadratic \blacktriangleright \text{discriminant}() \end{array}\right]$
	$solutions = \emptyset$
	IF $d < 0$ → RE *solutions*
	$\boxed{\neq}\left[\begin{array}{l} quadratic \blacktriangleright a \\ 0 \end{array}\right]$ → $solutions \overset{\pm}{=} \boxed{\div}\left[\begin{array}{l} -\sqrt{d} - quadratic \blacktriangleright b \\ 2 \times quadratic \blacktriangleright a \end{array}\right]$
	$solutions \overset{\pm}{=} \boxed{\div}\left[\begin{array}{l} \sqrt{d} - quadratic \blacktriangleright b \\ 2 \times quadratic \blacktriangleright a \end{array}\right]$
	RE *solutions*
	$\boxed{\neq}\left[\begin{array}{l} quadratic \blacktriangleright b \\ 0 \end{array}\right]$ → $solutions \overset{\pm}{=} \boxed{\div}\left[\begin{array}{l} -quadratic \blacktriangleright c \\ quadratic \blacktriangleright b \end{array}\right]$
	RE *solutions*
	RE *solutions*

4.1.12 Union_interval_angle

1. **Definition**

Union_interval_angle ◀ Array	
	method

2. **Method**

1	construct	

CO	*intervals_angle*
	$\boxed{=}\left[\begin{array}{l} intervals\_angle\_unioned \\ \emptyset \end{array}\right]$
	sublevel
	parent ▶ CO(*intervals_angle_unioned*)

FO $0 \leq i < |intervals\_angle|$

IF $\boxed{\equiv} \begin{bmatrix} intervals\_angle[i] \\ \emptyset \end{bmatrix}$ — CT

IF $|intervals\_angle[i]| \equiv 2\pi$ — $\boxed{=} \begin{bmatrix} intervals\_angle\_unioned \\ [intervals\_angle[i]] \end{bmatrix}$; BR

$\boxed{=} \begin{bmatrix} interval\_angle\_unioned \\ intervals\_angle[i] \end{bmatrix}$

$key\_unioned = 0$

1 1 FO $0 \leq i < |intervals\_angle\_unioned|$

IF $\boxed{\cap} \begin{bmatrix} interval\_angle\_unioned \\ intervals\_angle\_unioned[i] \end{bmatrix} \neq \emptyset$ — $interval\_angle\_unioned = \boxed{\cup} \begin{bmatrix} interval\_angle\_unioned \\ intervals\_angle\_unioned[i] \end{bmatrix}$; $\boxed{=} \begin{bmatrix} intervals\_angle\_unioned[i] \\ \emptyset \end{bmatrix}$

IF $\boxed{<} \begin{bmatrix} intervals\_angle\_unioned[i] \blacktriangleright start \\ intervals\_angle\_unioned \blacktriangleright start \end{bmatrix}$ — $\overset{+}{+} key\_unioned$

$intervals\_angle\_unioned = \text{array\_splice} \begin{pmatrix} intervals\_angle\_unioned \\ key\_unioned \\ 0 \\ [interval\_angle\_unioned] \end{pmatrix}$

2 complement — complement

complement

$\boxed{=} \begin{bmatrix} this\_complement \\ \textbf{Union\_interval\_angle} \end{bmatrix}$

IF $|this| \neq 0$

$interval\_angle = \textbf{Interval\_angle} \begin{pmatrix} 0 \\ 2\pi \\ \text{TR} \end{pmatrix}$

$\boxed{\overset{\cup}{=}} \begin{bmatrix} this\_complement \\ interval\_angle \end{bmatrix}$

RE $this\_complement$

FO $0 \leq i < |this|$

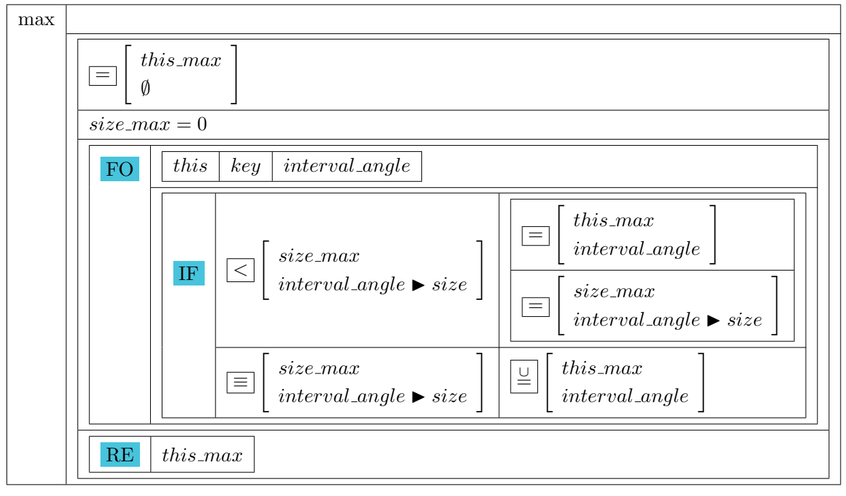
$\boxed{=} \begin{bmatrix} angle\_end\_i \\ this[i] \blacktriangleright \text{end}() \end{bmatrix}$

$interval\_angle = \textbf{Interval\_angle} \begin{pmatrix} angle\_end\_i \\ \boxed{-} \begin{bmatrix} this \blacktriangleright \begin{bmatrix} \text{interval\_angle}(i+1) \\ start \end{bmatrix} \\ angle\_end\_i \end{bmatrix} \end{pmatrix}$

$\boxed{\overset{\cup}{=}} \begin{bmatrix} this\_complement \\ interval\_angle \end{bmatrix}$

RE $this\_complement$

3 max — max size sub **Interval_angle**


max
= this_max ∅
size_max = 0
FO
this
key
interval_angle
IF
< size_max interval_angle ▶ size
= this_max interval_angle
= size_max interval_angle ▶ size
≡ size_max interval_angle ▶ size
∪≡ this_max interval_angle
RE
this_max