rpqvis

class vis.**Window**(*x*, *y*)

Bases: **object**

check_propagate_error(*id*)

Check if the insertion of a delete-min would cause another delete-min to hit the ceiling. This check is done by using an existing delete-min; it will ignore its current pair and propagate to the next lowest available insert.

Parameters:	**id** – tkinter ID of the delete-min line

clicked(*event*)

Respond to mouse clicks. In insert and delete mode, if the action is valid, a ray will be inserted or deleted.

Parameters:	**event** – tkinter event, holding location of mouse

delete(*t*, *y*)

Delete a delete-min or an insert. Nothing is done if the action is invalid.

Parameters:	• **t** – quantized t coordinate of the mouse click • **y** – quantized y coordinate of the mouse click

display_dot(*qt*, *qy*, *fill='black'*, *outline='black'*)

Display a dot, given “quantized” coordinates.

Parameters:	• **qt** – “quantized” t coordinate • **qy** – “quantized” y coordinate • **fill** – color of dot • **outline** – color of outline of dot

display_line(*qt1*, *qy1*, *qt2*, *qy2*, *fill='black'*)

Display a line, given “quantized” endpoint coordinates.

Parameters:	• **qt1** – “quantized” t coordinate of first endpoint • **qy1** – “quantized” y coordinate of first endpoint • **qt2** – “quantized” t coordinate of second endpoint • **qy2** – “quantized” y coordinate of second endpoint • **fill** – color of line

display_table(*t=99*)

Set table text to result of query at time t.

Parameters:	**t** – time of query

insert(*t*, *y*)

Insert a delete-min or an insert. Nothing is done if the action is invalid.

Parameters:	• **t** – quantized t coordinate of the mouse click • **y** – quantized y coordinate of the mouse click

motion(*event*)

Respond to mouse motion. In insert mode, there is a shadow ray that is blue if the insert is allowed and red otherwise. Motion in the bottom bar corresponds to an upward ray; otherwise, there is a horizontal ray. In delete mode, hovering over an existing line turns the lines red or blue, depending on if the deletion is allowed. In query mode, motion triggers a query of the current t location of the mouse.

Parameters:	**event** – tkinter event, holding location of mouse

popup()

Open the help popup.

quantize(*t*, *y*, *w*, *h*)

Map actual location onto a lattice; return the nearest lattice point.

Parameters:	• **t** – actual t coordinate • **y** – actual y coordinate • **w** – width of window • **h** – height of window

quantize_line(*c*, *w*, *h*)

Helper function; map line endpoints onto a lattice.

Parameters:	• **c** – actual line coordinates • **w** – width of window • **h** – height of window

query(*t*)

Return items in the priority queue at time t.

Parameters:	**t** – time of query

resize(*event*)

Shift and resize on-screen graphics if the window is resized.

Parameters:	**event** – tkinter event, holding new width and height

scale_dot(*id*)

Shift an existing dot, based on lattice coordinates.

Parameters:	**id** – tkinter ID number of dot

scale_line(*id*)

Resize and shift an existing line, based on lattice coordinates.

Parameters:	**id** – tkinter ID number of line

toggle(*event*, *key=None*)

Toggles between modes; resets some values and erases shadow rays.

toggle_delete()

Helper function; toggles to delete mode from a button press.

toggle_insert()

Helper function; toggles to insert mode from a button press.

toggle_query()

Helper function; toggles to query mode from a button press.

toggle_step()

Helper function; toggles stepthrough mode from a button press.

unquantize(*qt*, *qy*, *w*, *h*)

Map lattice point to actual location.

Parameters:	• **qt** – “quantized” t coordinate • **qy** – “quantized” y coordinate • **w** – width of window • **h** – height of window