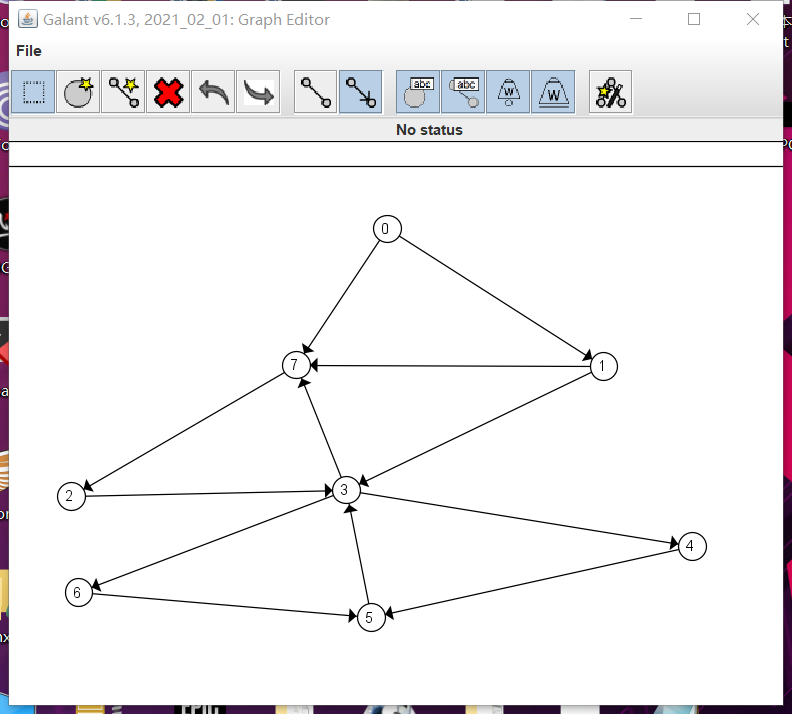
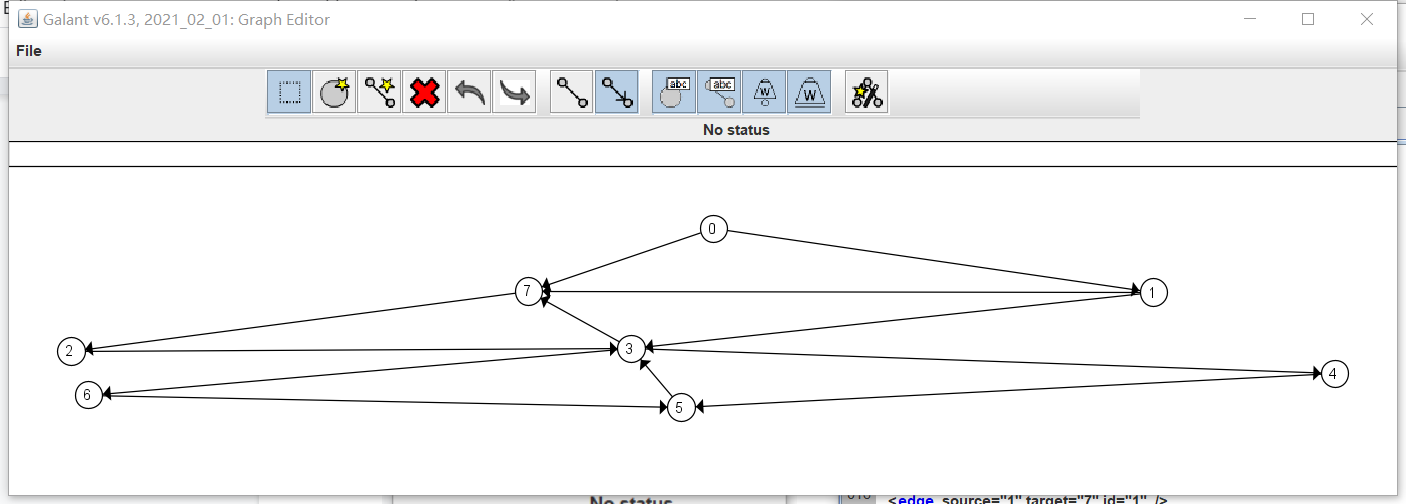
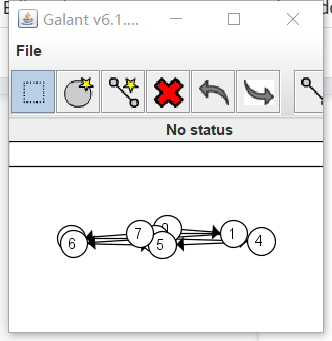
**Feature 1:** Auto-scale Graphical Elements with Window Resizing

With Galant being opened, please load a Non-Layered Graph first. For example, “unweighted\_8.graphml” is loaded below. (*Figure 1.1*)



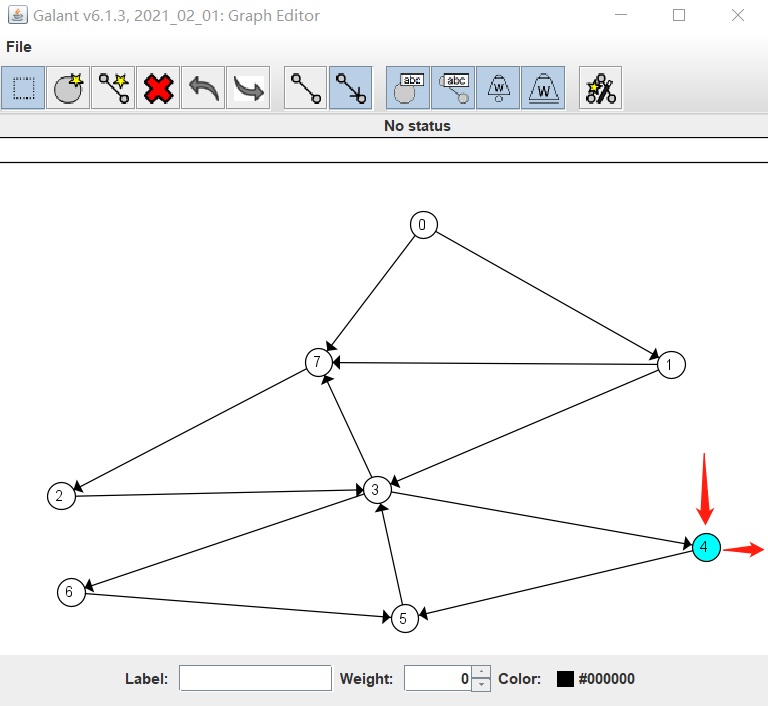
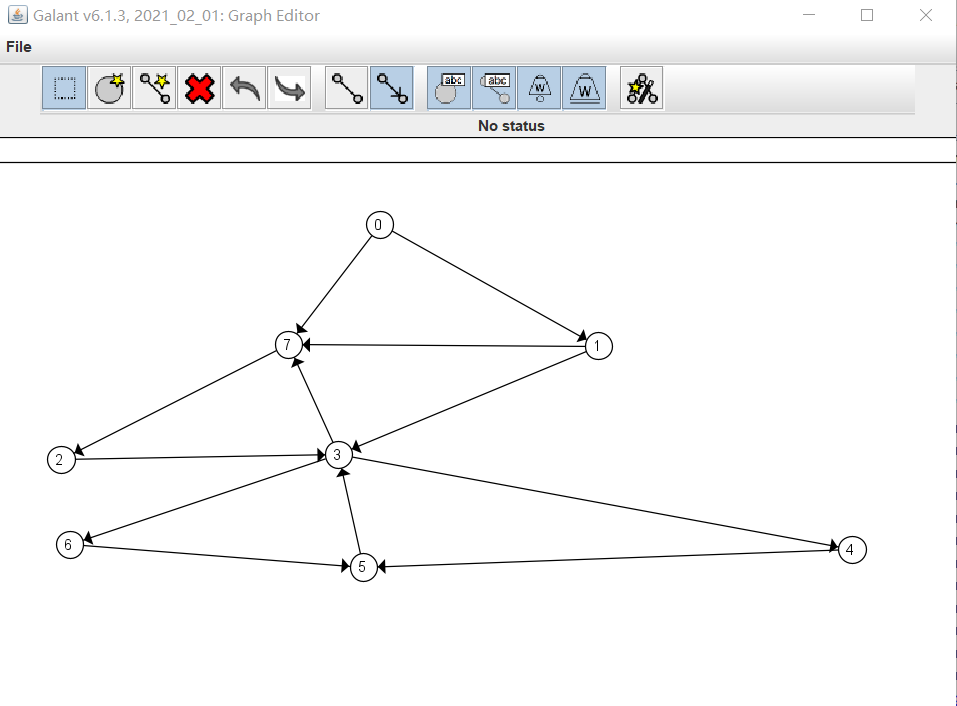
*Figure 1.1 Graph Loading*

You can then resize the window by dragging the border of the Graph Display Window. The graph will resize its graph elements based on the new window size automatically. (*Figure 1.2*)



*Figure 1.2 Window Resizing*

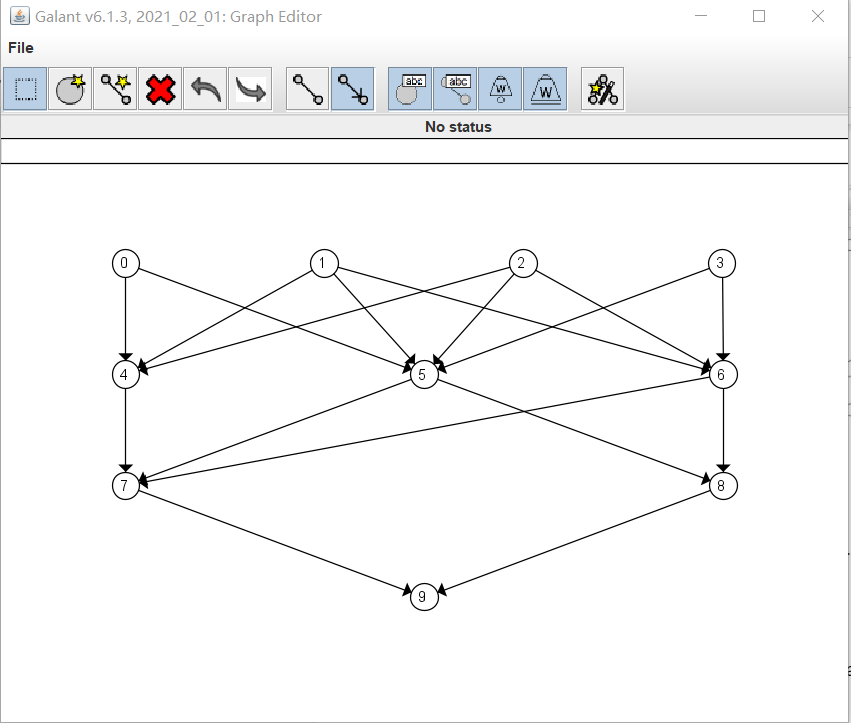
You can also drag a particular node towards the border of the Graph Display Window. As the dragged node gets closer to the border, the program should automatically expand the Graph Display Window towards that direction to adapt the movement you’ve made. (*Figure 1.3*)

*Figure 1.3 Boundary Check*

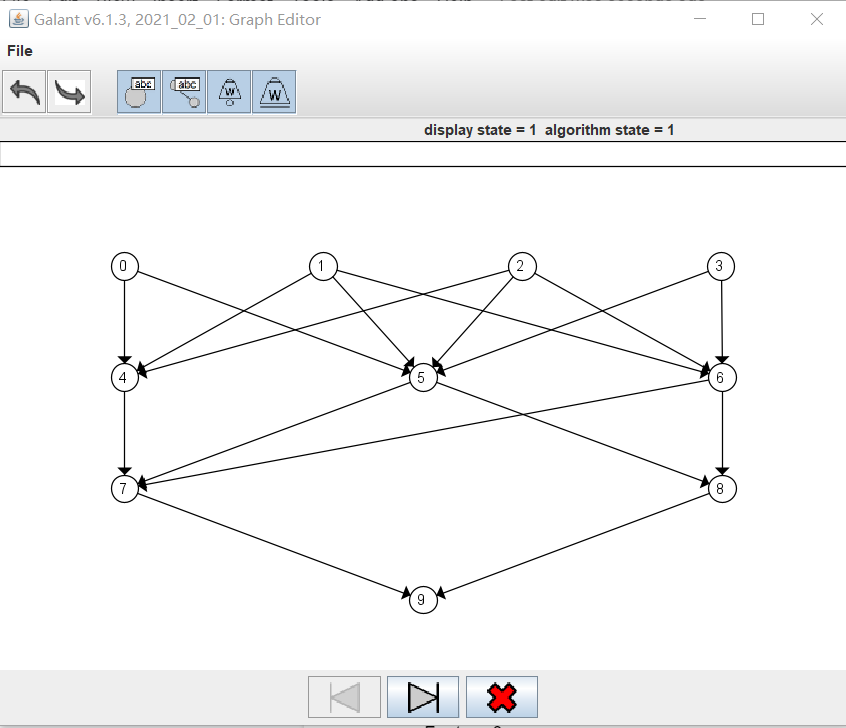
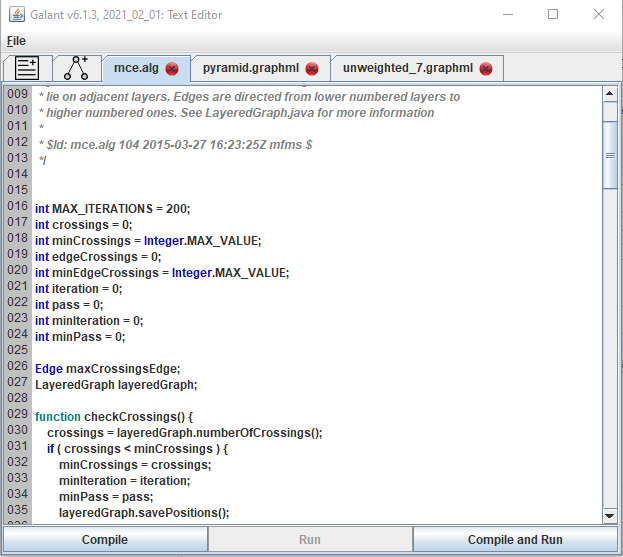
**Feature 2:** Enhancement of Editing Capability of Galant

Once you’ve opened Galant, please load a Layered Graph. For example, the “pyramid.graphml” can be used as an option. (*Figure 2.1*)



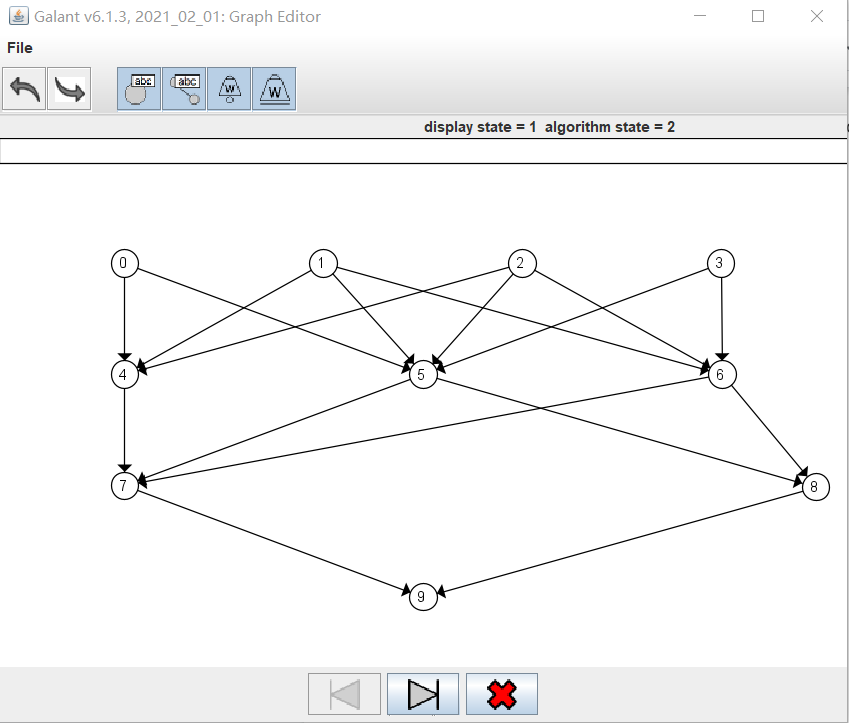
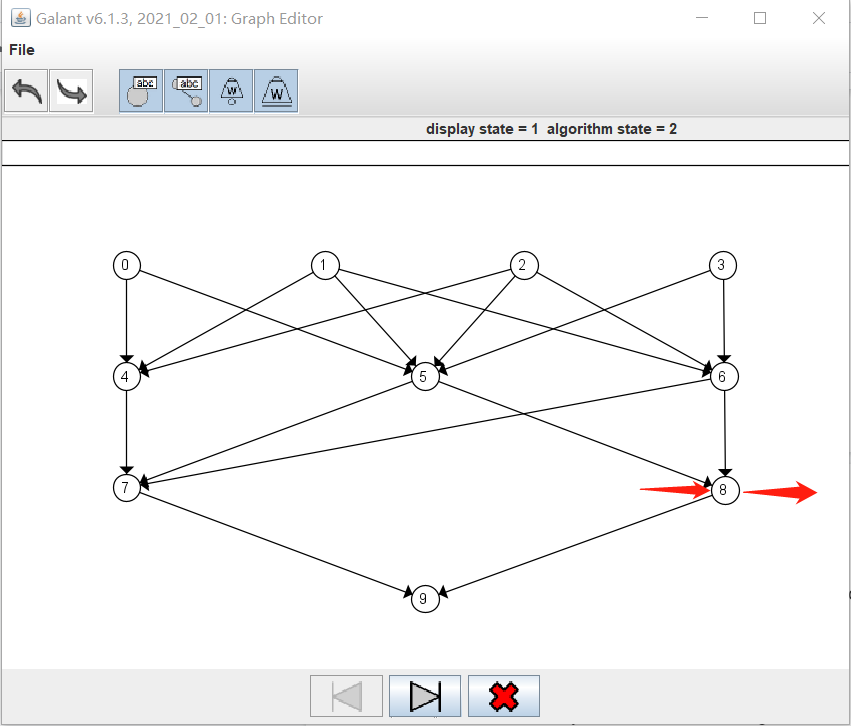
*Figure 2.1 Graph Loading*

Please load a Layered Graph Algorithm next in the editor window, for example, you can use “mce.alg.” Then click “compile and run.” (*Figure 2.2*)



*Figure 2.2 Algorithm Execution*

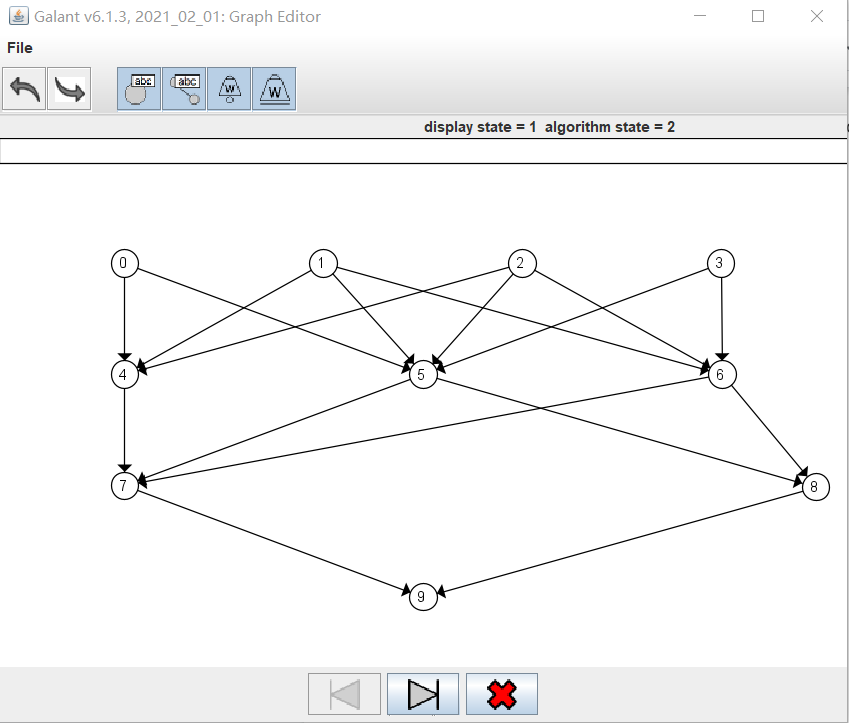
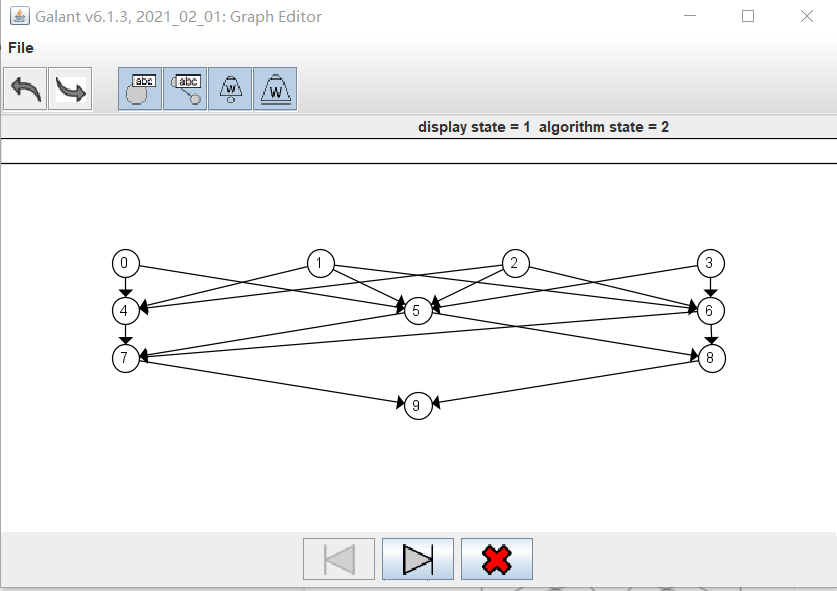
When the algorithm is running, the user can drag nodes inside the graph window. (*Figure 2.3*)



*Figure 2.3 Node Movement*

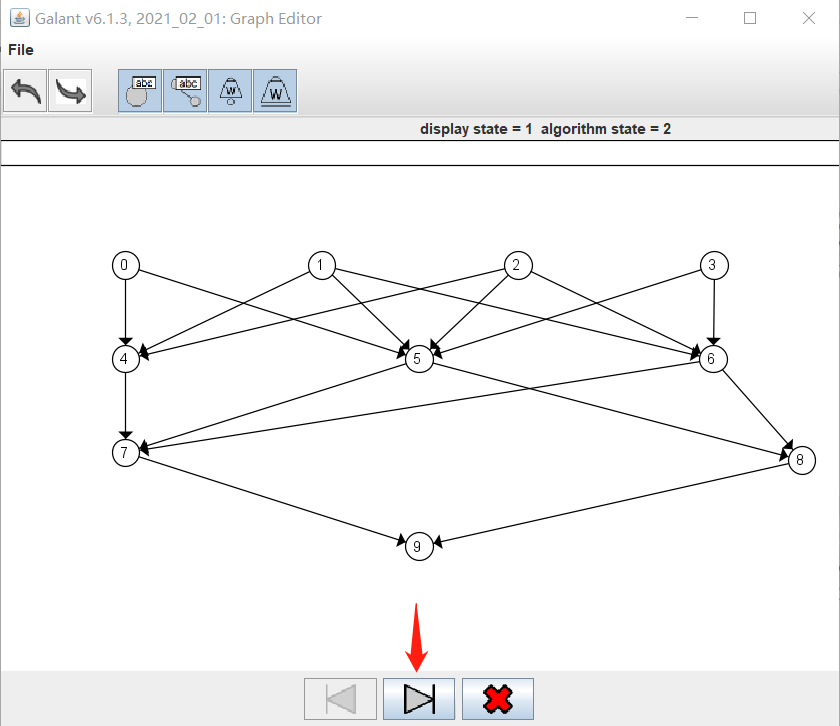
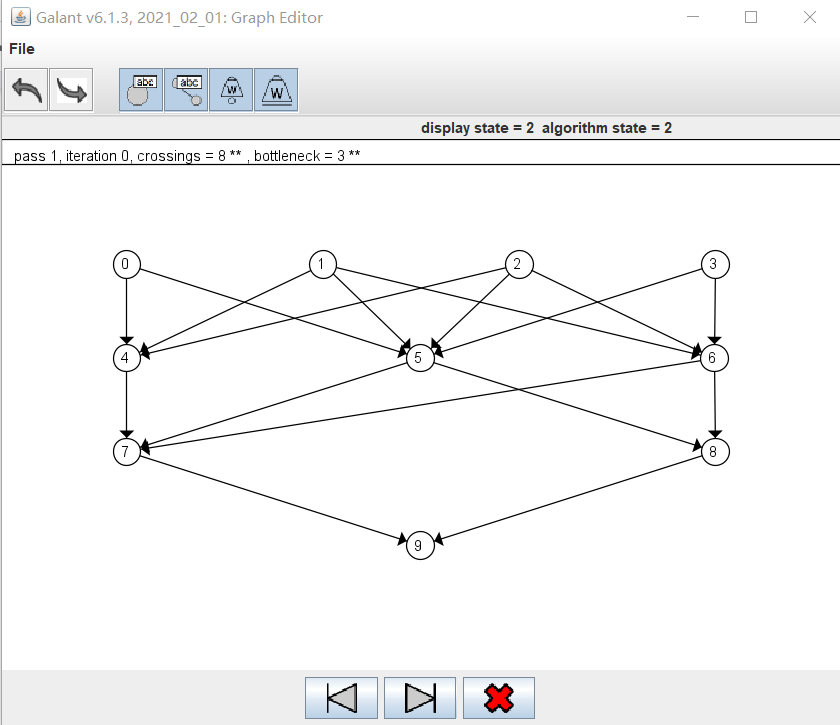
For LayeredGraph, the graph window will not expand when the node reaches the border, so the node may be dragged out of the window. But there are different ways to reposition all nodes back to their original positions.

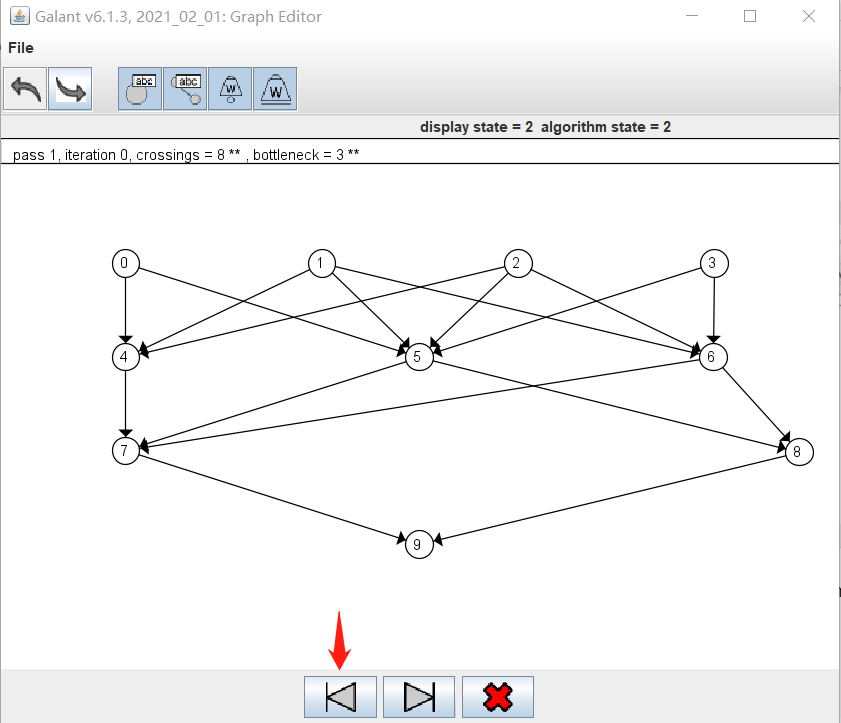
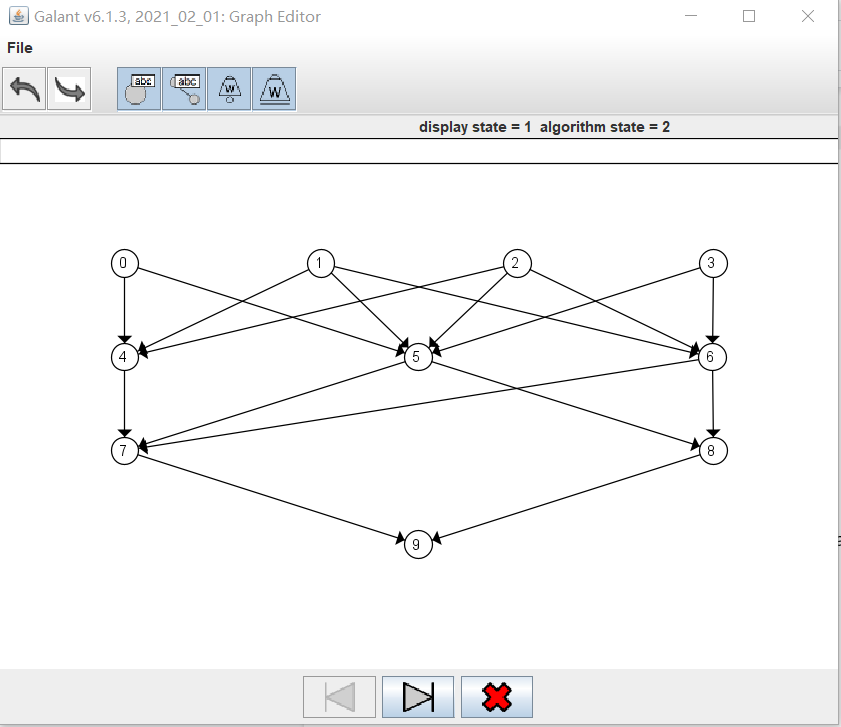
Method 1: Resizing the graph window: (*Figure 2.4*)

 → 

*Figure 2.4 Resizing to Trigger Reposition*

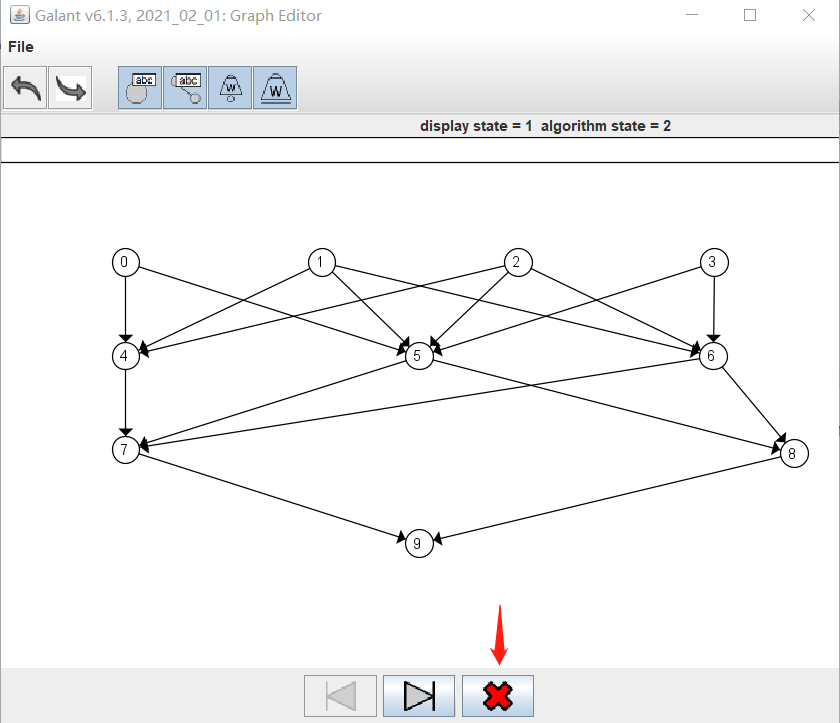
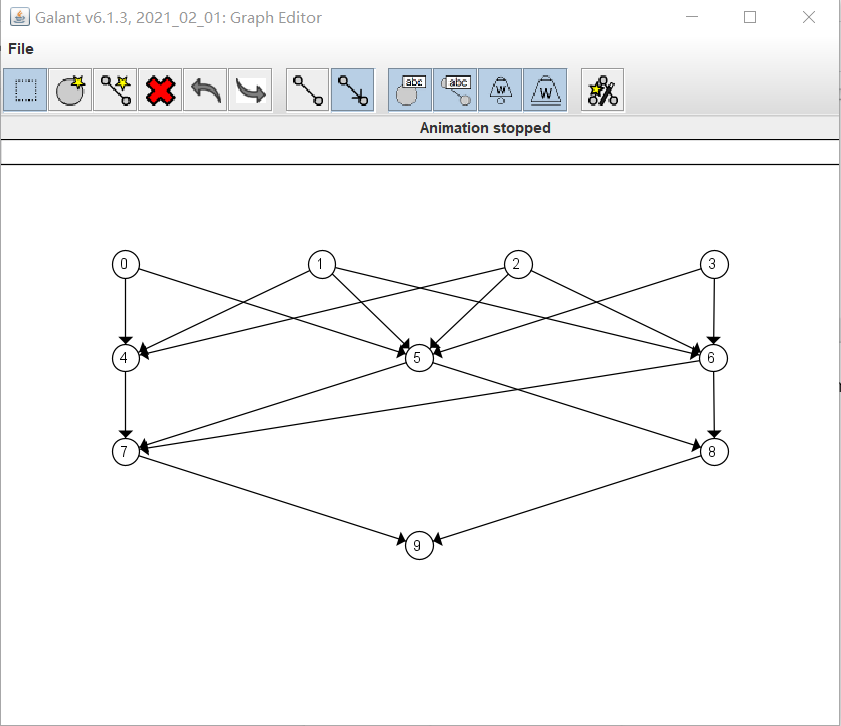
Method 2: Steping forward/backward during the execution of an algorithm: (*Figure 2.5*)

 → 

 → 

*Figure 2.5 Stepping Forwards/Backwards to Trigger Reposition*

Method 3: Exiting the animation mode (terminating algorithm): (*Figure 2.6*)

 → 

*Figure 2.6 Terminating Algorithm Execution*

Notice that you can’t resize the graph window/step forward/backward/exit animation mode without repositioning nodes.

Feature 3:

This feature is about simplifying the structure of code base and refactoring, it has no impact over the visual performance/user interface experience.