

## Chapter 14

# Hierarchical Modeling

Methods for model representations, hierarchical modeling, and programming implementations are considered in the exercises for this chapter.

### Exercises

- 14-1. Typical systems include architectural designs, electrical layouts, molecular models, and other scientific studies. For example, an architectural model requires components for buildings, while a molecular model requires atomic building blocks and their binding forces. And graphical representations for a model can be specified in terms of the general symbol shapes and the geometric primitives that could be used to display the symbol shapes. Other types of systems, such as those dealing with cosmology, mathematics, or economic trends, could also be considered.
- 14-2. A few simple furniture items, such as a rectangular desk and bookcase, in addition to the chair and worktable layouts shown in Fig. 14-11, can be displayed in the menu for this problem. And the floor plan can be a rectangle or other polygon shape. When a menu shape is selected, a copy of the shape can be created in some convenient screen location, and this copy can then be dragged to a position within the floor plan. Rotations could be applied interactively by selecting an object in the room and reorienting it by an amount that is specified by mouse movement in a particular direction.
- 14-3. Scaling amounts could be applied to a selected room item using menu selections or keyboard input. For example, items could be scaled in fixed increments using the up-down arrow keys.
- 14-4. For this exercise, modify the program from Exercise 14-2 so that several work-area floor plans are displayed along with a larger floor plan of the entire facility. Furniture items are then interactively placed into the individual work areas, and then the work areas are to be arranged within the overall facility floor plan, along with hallways and doors into the individual work areas.
- 14-5. The program in the previous exercise can be modified to include scaling as in Exercise 14-3.

- 14-6. For this exercise, develop a program to display a menu of logic shapes and to create an instance of any item selected from the menu.
- 14-7. The routines for this exercise are similar to those for Exercise 14-2, but in this case the menu displays logic symbols (Exercise 14-6). As an instance of a selected menu item is placed into the circuit layout, straight-line segments are drawn to connect the new item to existing items in the circuit. Interactive input can be used to select the two endpoint positions for each line connector.
- 14-8. A general description of the routines needed for module editing can first be devised. Then the individual routines could be implemented for a particular system, such as the logic-circuit design in the previous exercise.
- 14-9. Input for this routine is a coordinate position that has been interactively selected within a screen display. The input position is then checked against the coordinate extents of the displayed objects, using the picking methods discussed in Sections 11-2 and 11-6. If these checks fail to identify the pick object, an error message can be displayed. Otherwise, the picked object is deleted from the display. Special considerations may need to be applied to the deletion operations, depending on the particular features of a system model. For example, when an element is deleted from a circuit design, the circuit connections need to be revised.
- 14-10. The procedures for this exercise are an extension of those for the previous exercise. In this case, the input position is used to identify a group of elements. Thus, the picking operations need to check the module definitions, as well as the coordinate extents of individual elements. Menu or keyboard options can be used to identify whether the picking operation is to be used to choose an individual element of the display or a module.
- 14-11. This routine is a modification of the procedures in the previous exercise, where a selected module can be temporarily deleted, then returned to the display. Thus the details for all deletion operations need to be saved so that the model can be restored if a module is returned to the display.
- 14-12. Highlighting operations can be applied to an individual element in the display or to a group of elements. As an activated mouse moves through a display, picking operations are implemented and picked objects can change color or be enclosed within a color wire-frame rectangle.
- 14-13. A blinking object (or group of objects) can be displayed by alternately making the geometric structures invisible then visible, using a time delay between the two operations. Geometric primitives are invisible, when they are displayed in the background color.