```
                              Math.log(sandpile.distribution[s]*1.0/sandpile.numberOfGrains));
      }
    }
    plotFrame.render();
  }

  public void reset() {
    control.setValue("L", 10);
    enableStepsPerDisplay(true);
  }

  public void resetAverages() {
    sandpile.resetAverages();
  }

  public static void main(String[] args) {
    SimulationControl control = SimulationControl.createApp(new SandpileApp());
    control.addButton("resetAverages", "resetAverages");
  }
}
```

Problem 14.7. A two-dimensional sandpile model

a. Use the classes `Sandpile` and `SandpileApp` to simulate a two-dimensional sandpile with linear dimension L. Run the simulation with $L = 10$, and stop it once toppling starts to occur. When this behavior occurs, black cells (with four grains) will momentarily appear. Use the Step button to watch individual toppling events, and obtain a qualitative sense of the dynamics of the sandpile model.

b. Comment out the `height.render()` statements in `Sandpile`, and add a statement to `SandPileApp` so that the number of grains added to the system is displayed. (The number of grains added is a measure of the number of configurations that are included in the various averages.) Now you will not be able to see individual toppling events, but you can more quickly collect data on the toppling distribution, the frequency of the number of sites that topple when a grain is added. The program outputs a log-log plot of the distribution. Estimate the slope of the log-log distribution from the part of the plot that is linear and thus determine the power law exponent α. Reset the averages and repeat your calculation to obtain another estimate of α. If your two estimates of α are within a few percent, you have added enough grains of sand. Compute α for $L = 10$, 20, 40, and 80. As you make the lattice size larger, the range over which the log-log plot is linear should increase. Explain why the plot is not linear for large values of the number of toppled sites.

Of course, the model of a sandpile in Problem 14.7 is over simplified. Laboratory experiments indicate that real sandpiles show power law behavior if the piles are small, but that larger sandpiles do not (see Jaeger et al.).

Earthquakes. The empirical Gutenberg-Richter law for $N(E)$, the number of earthquakes with energy release E, is consistent with power law behavior:

$$N(E) \sim E^{-b}, \tag{14.3}$$

with $b \approx 1$. The magnitude of earthquakes on the Richter scale is approximately the logarithm of the energy release. This power law behavior does not necessarily hold for individual fault systems, but holds reasonably accurately when all fault systems are considered. One implication of the power law dependence in (14.3) is that there is nothing special about large earthquakes. In Problems 14.8 and 14.9 and Project 14.26 we explore some models of earthquake models.

Given the long time scales between earthquakes, there is considerable interest in simulating models of earthquakes. The Burridge-Knopoff model considered in Project 14.26 consists of a system of coupled masses in contact with a rough surface. The masses are subjected to static and dynamic friction forces due to the surface, and also are pulled by an external force corresponding to slow tectonic plate motion. The major difficulty with this model is that the numerical solution of the corresponding equations of motion is computationally intensive. For this reason we consider several cellular automaton models that retain some of the basic physics of the Burridge-Knopoff model.

Problem 14.8. A simple earthquake model

Define the real variable $F(i, j)$ on a square lattice, where F represents the force or stress on the block at position (i, j). The initial state of the lattice at time $t = 0$ is found by assigning small random values to $F(i, j)$. The lattice is updated according to the following rules:

(i) Increase F at every site by a small amount ΔF, for example, $\Delta F = 10^{-3}$, and increase the time t by 1. This increase represents the effect of the driving force due to the slow motion of the tectonic plate.

(ii) Check if $F(i, j)$ is greater than F_c, the threshold value of the force. If not, the system is stable and step 1 is repeated. If the system is unstable, go to step 3. Choose $F_c = 4$ for convenience.

(iii) The release of stress due to the slippage of a block is represented by letting $F(i, j) = F(i, j) - F_c$. The transfer of stress is represented by updating the stress at the sites of the four neighbors at $(i, j \pm 1)$ and $(i \pm 1, j)$: $F \to F + 1$. Periodic boundary conditions are not used.

These rules are equivalent to the Bak-Tang-Wiesenfeld model. What is the relation of this model to the sandpile model considered in Problem 14.7?

As an example, choose $L = 10$. Do the simulation and show that the system eventually comes to a statistically stationary state, where the average value of the stress at each site stops growing. Monitor $N(s)$, the number of earthquakes of size s, where s is the total number of sites (blocks) that are affected by the instability. Then consider $L = 30$ and repeat your simulations. Are your results for $N(s)$ consistent with scaling?

Problem 14.9. A dissipative earthquake model

The Bak-Tang-Wiesenfeld earthquake model discussed in Problem 14.8 displays power law scaling due to the inherent conservation of the dynamical variable, the stress. It is easy to modify the model so that the stress is not conserved and the model is more realistic. The Rundle-Jackson-Brown/Olami-Feder-Christensen model of a earthquake fault is a simple example of such a nonconservative system.

a. Modify the toppling rule in Problem 14.8 so that when the stress on site (i,j) exceeds F_c, not all the excess stress is given to the neighbors. In particular, assume that when site (i,j) topples, $F(i,j)$ is reduced to the residual stress $F_r(i,j)$. The amount $\alpha(F_{ij} - F_r)$ is dissipated leaving $(F_{ij} - F_r)(1-\alpha)$ to be distributed equally to the neighbors. If $\alpha = 0$, the model is equivalent to the model considered in Problem 14.8. Choose $\alpha = 0.2$ and determine if $N(s)$ exhibits power law scaling. For simplicity, choose $F_c = 4$ and $F_r = 1$ (see Grassberger).

b. Make the model more realistic by adding a small amount of noise to F_r so that F_r is uniformly distributed between $1-\delta, 1+\delta$ with $\delta = 0.05$. Also run the model in what is called the "zero-velocity limit" by finding the site with the maximum stress $F_{\max}$ and then increasing the stress on all sites by $F_c - F_{\max}$ so that only one site initially becomes unstable. Determine $N(s)$ and see if your results differ from what you found in part (a). Do you still observe power law scaling?

c. The model can be made more realistic still by assuming that the interaction between the blocks is long range due to the existence of elastic forces. Distribute the excess stress equally to all z neighbors that are within a distance of radius R of an unstable site. Each of the z neighbors receives a stress equal to $(F_{ij} - F_r)(1-\alpha)/z$. First choose $R = 3$ and see if the qualitative behavior of $N(s)$ changes as R becomes larger. Lattices with $L \geq 256$ are typically considered with $R \simeq 30$ (see Rundle et al.).

The behavior of some other simple models of natural phenomena is explored in the following.

Problem 14.10. Forest fire model

a. Consider the following model of the spread of a forest fire. Suppose that at $t = 0$ the $L \times L$ sites of a square lattice either have a tree or are empty with probability p and $1-p$ respectively. The sites that have a tree are on fire with probability f. At each iteration an empty site grows a tree with probability g, a tree that has a nearest neighbor site on fire catches fire, and a site that is already on fire dies and becomes empty. This model is an example of a probabilistic cellular automaton. Write a program to simulate this model and color code the three types of sites. Use periodic boundary conditions.

b. Choose $L \geq 30$ and determine the values of g for which the forest maintains fires indefinitely. Note that as long as $g > 0$, new trees will always grow.

c. Use the value of g that you found in part (b) and compute the distribution of the number of sites s_f on fire. If the distribution is critical, determine the exponent α that characterizes this distribution. Also compute the distribution for the number of trees, s_t. Is there any relation between these two distributions?

d.* To obtain reliable results it is frequently necessary to average over many initial configurations. However, the behavior of many systems is independent of the initial configuration and averaging over many initial configurations is unnecessary. This latter possibility is called *self-averaging*. Repeat parts (b) and (c), but average your results over ten initial configurations. Is this forest fire model self-averaging?

Problem 14.11. Another forest fire model