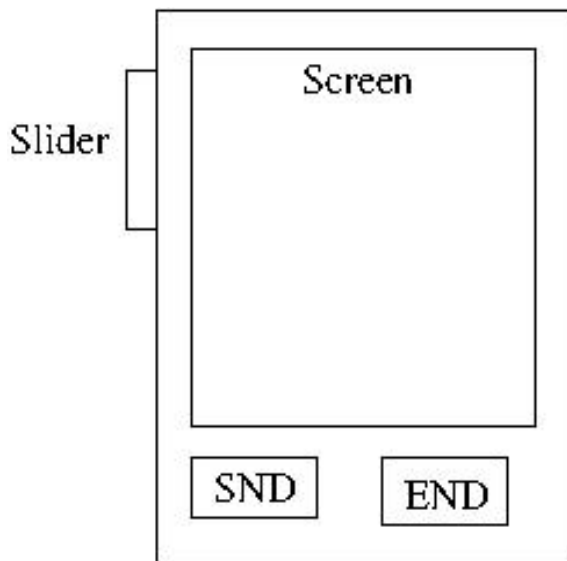# CS 347 State design pattern Test

Consider the following "dumbphone" design:

The dumbphone consists of only a screen and three keys: SND, END, and a slider that can be pushed up or down. The keys function as follows:

SND key – When the phone is in standby mode, this key causes a transition into call mode.

END key – When in call mode or application mode (i.e., running an application), this key causes the phone to transition into standby mode.

Slider – When in standby mode, this key raises or lowers ringer volume. When in call mode, this key raises or lowers voice volume. When in application mode, this key scrolls the display up/down.

Write code to operate the dumbphone.
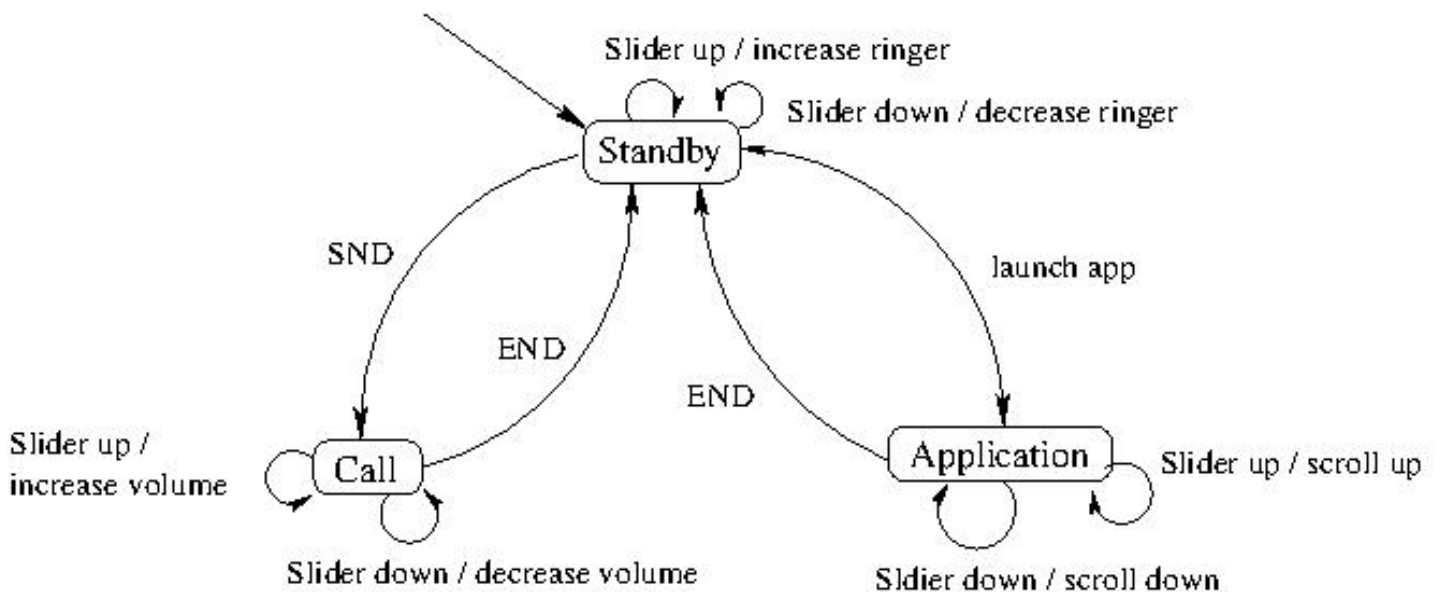
Your code should use the State design pattern. Recall that steps for using the State design pattern are:

1. Identify the states. Assume that when the phone is in standby mode, an application can be launched—and the phone transitioned into application mode—by some mechanism that doesn't use a key; e.g., perhaps using some control on the screen.
2. Besides objects that represent states, there will be a "context" object
3. Recall that a decision choice that is not specified by the pattern is whether state transitions are initiated by the context object or by the state objects.