1. Circle all of the following that are true about the Model-View-Controller pattern:
   (a) The model is written to work only with one or two kinds of views. Adding a new kind of view always requires adding new code to the model. **False.** *The idea of MVC is you can add views or controllers without changing the model.*
   (b) A model can have at most one view associated with it. **False.**
   (c) A controller sends messages to the model causing it to change its state. **True**
   (d) Different kinds of views can be associated with a given model, all displaying information in different ways. **True** *(LunarLander is an example)*
   (e) The view must constantly query the model, seeing whether anything has changed. **False.** *The view doesn't need to constantly query the model; only when the model notifies of a change must it query.*
   (f) Each Swing JCheckbox has a model associated with it. **True**

2. Circle all of the following that are true:
   (a) Generic classes have a type parameter. **True**
   (b) An `ArrayList<Critter>` cannot store an Ant object. **False.** *It can store any subclass of Critter*
   (c) An `ArrayList<Critter>` cannot store a String. **True**
   (d) The following is legal: **False.** *Otherwise, you could add objects to al*
       ```java
       ArrayList<Object> al = new ArrayList<String>;
       ```
   (e) The following is legal: **False.** *Otherwise, you could retrieve a non-String from as*
       ```java
       ArrayList<String> as = new ArrayList<Object>;
       ```
   (f) Declaring `al` of type `ArrayList<Critter>` tells the compiler that `al` will only store Critters or some subclass. **True**