

## 4 A Creative Project

We would like you now to use what you have learned and see whether it helps you in designing a new independent project. Our `javilib` libraries allow you to design a simple interactive game with very little work.

A game consist of several different objects. The object move either on each tick of the clock, or in response to the keys (typically the arrow keys). There may be other changes in the game object over the time or in response to the key events (x key launches a shot, an animal gets hungrier as the time goes on, ...). The objects interact in some predefined manner. Finally, something (the state of an object, the interaction between objects) triggers the end of the game.

The simple example of a `BlobWorld` shows you what is needed to design a game and run it.

- A. Design a simple game with no more than three kinds of interacting objects.
- B. For each object that will be used in the game do the following:
  - (a) Describe briefly its behavior during the game: does it change with the clock tick?, does in respond to key events?, does it interact with another object in the game?
  - (b) Identify the essential information you will need to keep track of as the *World* scene changes. Design a class to represent this information and make examples of data, especially those at the beginning of the game, or in any expected unusual situations during the game.
  - (c) Design the class `GameWorld` that includes all objects involved in the game.
  - (d) Make examples of the initial `GameWorld` and a couple of intermediate *worlds* you expect to see in the game.
  - (e) Design a picture that will represent each of the objects in your game. A picture is composed of disks, circles (outlines), rectangles, lines and text in six possible colors: red, blue, green yellow, white, or black.
  - (f) Design the `draw` method for each object in the game and for the whole `GameWorld`. The methods for each object consume

the instance of the `Canvas` on which the object image is to be drawn.

- (g) Design the method `onKeyEvent` that produces a new `World` in response to some key events.
- (h) Design the method `onTick` that produces a new `World` in response to the next tick of the clock

*Note:* Beware. There will be a test! We will examine your code and the code of your peers later in the week, looking for the code quality and the quality of the test design.