EnClojure – A 2D Adventure Game

Yoo Min Cha || Michael Jannino

Problem:

The problem we are tasked with is to create a 2d game environment using functional programming and development that is engaging, fun, and works. Rather than iteratively defining a game world, we must program in a reaction-based way. This is interesting because it provides an opportunity to use a strength of lisp-based languages—object persistence. This combined with a REPL results in an extremely fast development environment as changes are reflected immediately. Tweaking will be trivial, but at the cost of iteratively walking through a sequence of events. Applying functions to a set of data rather than feeding data to a function will present a very interesting way of thinking about a game; instead of an environment of data that can be passed into functions, we will have an environment of objects and functions that interplay with each other.

Problem Analysis:

OPL has done a good job preparing us for tackling this problem. The most obvious and most important of the concepts that we've learned is object orientation. A game environment essentially requires object orientation; conceptually, the world is full of objects that are able to do certain things, and certain objects are subsets of other objects. Functions as first-class objects is arguably just as important; 'events' in the world can have a more sophisticated relationship with objects in this way. Object interaction and multiple dataset manipulation is going to be the 'backbone' of how objects behave, and a great way to do this is to feed objects functions rather than feed functions objects. Data abstraction will also be an extremely important facet of our project, as it goes hand-in-hand with object-orientation. This will make optimizations a lot less bulky and time-consuming to implement, as we will have levels of data and object handling that do not depend on the specific implementation of certain algorithms.

Data Set/Source Materials:

Our environment will be using data that we generate ourselves through actual art, algorithms, and the resulting environment. Loading in the art we create and assembling them on screen will constitute the environment, and the way the objects behave will be the result of the algorithms and related manipulation. Our plan for processing the game environment data every second is by examining existing game engines, some existing AI constructs, and attempting to apply the concepts we take away from that experience to our own world. We will most likely be using the LightTable text editor for the robust Insta-REPL, and will be exploring the Unity3D/2D engine for game environment development.

Deliverable and Demonstration:

Our deliverable is a fully functional, engaging 2D game. By nature, games are interactive, and if we get far enough along in development of the game for the project we plan to release it for individual play. Our goal with this is to make something that lasts longer than our time in the class. For our demonstration, we can play through the game and show the various functions that work with each other and how they translate to the game environment.

Evaluation of Results:

The most quantifiable analysis of the game we create is frame rate and how well objects interact with each other. There will be definite bugs due to the short development time, but some optimizations ideally will have taken place. "Fun" factor is the biggest goal of the game, but also the least quantitative result; if we have a working game that can engage at least one person, we will have been successful.

Work Plan/Schedule:

We will have three deliverables. The first deliverable on **Mon 11/24** will be a successful printing of an object on the screen with some degree of manipulation available. The second deliverable on **Mon 12/1** will be character manipulation in the environment we develop, and core implementation of the AI. The last deliverable, as the final project on **12/8**, will be a fully-functional game with a clear intent and solvable problems presented to the player. Below is a breakdown of what each of us will do for each deliverable.

	What I will do: Yoo Min Cha	*What my partner will do: Michael Jannino*
11/24	Develop basic movement functions, and interface with printed objects.	Solidify printing things to the screen and create a basic character by importing art.
12/1	Develop algorithms for AI and completely flesh out character movement.	Create full environment, and implement advanced object printing to screen. Interface objects with algorithms.
12/8	Full flesh out event handling for entire game and ensure all objects interface correctly during gameplay. Bug fixing.	Design levels of the game, and provide challenges for player while finishing development of enemies/characters/environment. Bug fixing.