

Software Development I

Game Project v0.2

Overview

This semester we are building an interactive adventure game in Java in the object-oriented style.

For the next version of your project, you will build upon version 0.1 by adding a breadcrumb trail that the user drops with each move. This will include two new Java Classes.

Deliverables

- This version must include all features from the Version 0.1 game framework as described in the Game Project v0.1 Requirements document.
 - Any code cleanup, re-design, added content, etc. is also welcome.
- As the she navigates the World, the Player leaves a breadcrumb trail that can be followed in reverse.
 - With each move, *automatically* drop another breadcrumb in the new location.
- Add a new **Backtrack** command (character shown in **bold**) and corresponding Player *method*.
 - Player can specify the number of steps given as a second token, as in: **b 3**
 - Each step backtracking moves the Player to her previous location and removes (“picks up”) the previously dropped breadcrumb.
 - Attempting to backtrack when there are no more breadcrumbs should not cause a crash, but instead should notify the player that she has reached the beginning.
- Add new classes for the following game objects:
 - BreadCrumb – this is like a node in a linked list.
 - BreadCrumbTrail – this is like a stack consisting of BreadCrumbs.
- After each game update, the rendered output should include one additional bit of information:
 - number of steps (i.e., breadcrumbs) from the start

Important: You must write BreadCrumb and BreadCrumbTrail from scratch without using *java.util* classes. Refer to SimpleLinkedList under Resources > Source Code for guidance. Note that stacks are just lists restricted to add/get/remove at one end only.

Source Code

Your program design should clearly use separate objects for distinct concerns. Your code must be consistently formatted and demonstrate accepted best practices.

Submitting

Create a new Branch of your BitBucket repository called **v0.2**. Push changes frequently to this new branch with concise and relevant commit messages. Open an Issue if you need any help along the way – I generally reply quickly.

Send me a Pull Request for your v0.2 branch before the iLearn due date, and then under iLearn Assignments Attach and Submit an executable JAR file of your game.

Once graded, merge your branch it into the *master*.