Andrew Berger

Changelog

Pellissippi Garden Glanner

Date Changes

---------------------------------------------------------------

6-15-2017 Created UML Diagram and object map concepts

10-26-2017 Made a program in c++ to detect object overlap

Compiled and tested overlap binary with rectangles : )

10-28-2017 Implemented overlap detection for circles

Integrated Clipper API for overlap detection of any polygon

Revised overlap theory to start with circles or rectangles,

then deep test with Clipper if necessary.

Created an algorithm for “Polygonized ellipses”

in function polygonFromEllipse()

Implemented Konva API to draw polygons in JavaScript

Tested collision with every instance of two shapes! Success!

11-22-2017 Coded ellipse algorithm.

Implemented command line arguments to input polygons

Coded algorithm to draw circles around shapes to test overlap first

11-23-2017 Organized shape inheritance with magic pointers

Created ...from string functions to build shapes from input arguments

11-24-2017 Coded overlap algorithm to detect the shape’s overlap as circles, then as polygons

if necessary. Integrating overlap algorithms and functions to detect any two polygons.

Fixed a bug in the code that parses shape data from command line arguments

Narrowing down the code to get a clean binary!!!!!!!

12-28-2017 Completed Overlap Binary saved to test server.

Wrote a script that utilizes the binary, executing it from php and returning the overlap

result. Solved a glitch regarding duel libstd file includes that resulted from the default

Ubuntu xampp server settings.

12-29-2017 Created subclasses for all garden shapes and garden elements, and scaled the Konva JS

grid to match design pattern.

12-30-2017 Designed inheritance pattern for file IO.

Created the Garden class that contains instances of Garden elements.

12-31-2017 Implemented the Shapes and GardenElements classes and constructors.

Wrote the ClipDetector Class that allows shapes to be passed into binary for overlap

check, including algorithms for shape to string conversion.

Wrote the FileIO class that allows users to upload and download save files for the

garden, including a fix for XML injection vulnerabilites.

Implemented drag-and-drop functionality for shapes, tested event handlers.

HAPPY NEW YEAR!

1-17-2018 Created ajax functionality, made Visual Garden class, defined constants

1-19-2018 Began the file IO procedure to read and write to xml files

Began correcting the structure of garden elements

Added functionality for file upload

Devised file upload and Visual Garden Hierarchy

1-20-2018 Create a basic sample .xml file and uploaded it, parsed it, and saved it to the user’s

downloads.

Created an actual sample of a .garden save file that comprehensively detailed the

program save file structure. (See sample.garden in the Plans folder)

tried using the usda plant database api https://plantsdb.xyz/search, yet common name

search failed with names like "Green Beans" or "Bell Peppers" due to common

misnomers. Implemented more practical approach of simple\_html\_dom and google

image parse, using the common name that the user types to fetch an image and displayed

the typed name verbatim. Later corrected the search to only hit Burpee.com to avoid

pulling up images that were non plant related.

1-21-2018 Made the file IO load feature that completely breaks a .garden save file into php garden

objects. The VisualGarden class will draw these objects to the screen in Javascript.