Rose-Hulman Institute of Technology

Project Proposal

Programming Language Paradigms

James Gibson, Kieran Groble, Jenna Wohlpart

January 27, 2017

Proposal

Our final project will be to implement the Bently-Ottmann Algorithm to detect line segment intersections. A large part of the project will be to implement the supporting data structures on our own using dependent types in Haskell. One major reason that Haskell is well suited for the problem is Haskell's dependent types. With dependent types we can enforce the validity of the red-black tree which will be backing our algorithm. Another reason that Haskell will be well suited is that our project is clearly a function which takes in line segments and returns the crossing segments. We will not have to fight with its lack of state as a result. In addition, we will see what we can learn from other parts of the language. In particular, we hope to explore the type classes which are missing from Elm.

Because the world is not purely functional, we will also include file I/O and a visualization of the intersecting line segments. This will give us a more "real world" view of Haskell as a general purpose language.

Lastly, we will implement a number of unit tests to verify that our data structures and algorithm work as expected. This will allow us to see some of the benefits of purely functional languages, as our unit tests will not need to include any state management.

Many parts of the project will be coded in trios as we feel that they are all critical to our understanding. We do not feel the red-black tree should be done alone by one person as we would each like to use the dependent types. We do not feel the algorithm should be broken apart as we would each like to understand the final algorithm, although focus can be split by the person that wrote the corresponding supporting data structure. The rest will be broken apart as necessary.

Red-Black Trees		
Lookup and Structure	10	All
Insertion	5	Kieran
Removal	5	Coleman
Priority Queue		
Works	10	Jenna
Bently-Ottmann Algorithm		
Works	5	All
File I/O and Drawing	5	Coleman and Kieran
Tests		
Completed	5	All
Extra		
Completed	5	???