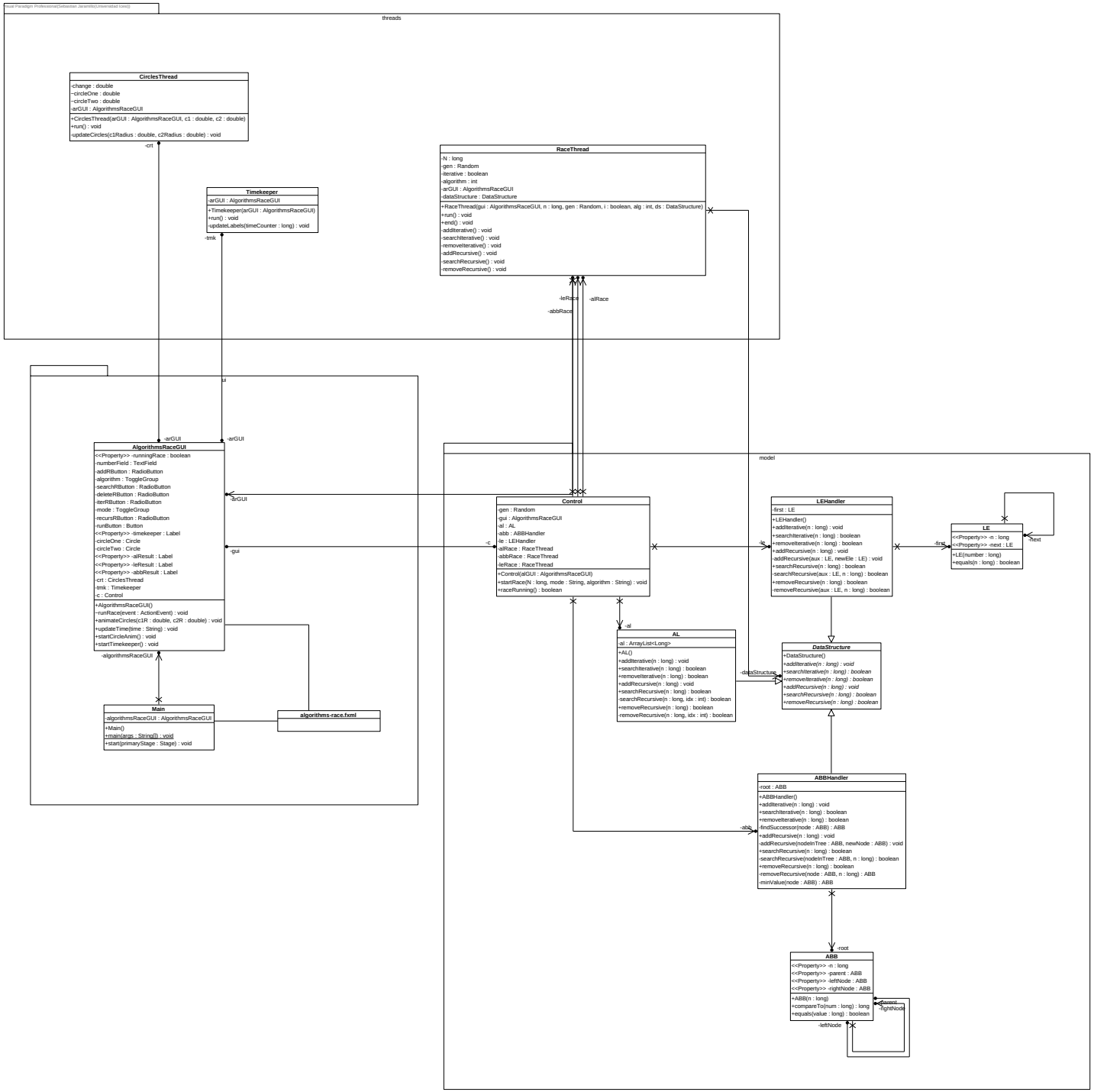
Visual Paradigm Professional(Sebastian Jaramillo(Universidad Icesi))
threads
CirclesThread
-change : double
−circleOne : double
−circleTwo : double
-arGUI : AlgorithmsRaceGUI
+CirclesThread(arGUI : AlgorithmsRaceGUI, c1 : double, c2 : double)
+run() : void
-updateCircles(c1Radius : double, c2Radius : double) : void
-crt
Timekeeper
-arGUI : AlgorithmsRaceGUI
+Timekeeper(arGUI : AlgorithmsRaceGUI)
+run() : void
-updateLabels(timeCounter : long) : void
-tmk
RaceThread
-N : long
-gen : Random
-iterative : boolean
-algorithm : int
-arGUI : AlgorithmsRaceGUI
-dataStructure : DataStructure
+RaceThread(gui : AlgorithmsRaceGUI, n : long, gen : Random, i : boolean, alg : int, ds : DataStructure)
+run() : void
+end() : void
-addIterative() : void
-searchIterative() : void
-removeIterative() : void
-addRecursive() : void
-searchRecursive() : void
-removeRecursive() : void
-leRace
-abbRace
-alRace
ui
-arGUI
-arGUI
AlgorithmsRaceGUI
<<Property>> -runningRace : boolean
-numberField : TextField
-addRButton : RadioButton
-algorithm : ToggleGroup
-searchRButton : RadioButton
-deleteRButton : RadioButton
-iterRButton : RadioButton
-mode : ToggleGroup
-recursRButton : RadioButton
-runButton : Button
<<Property>> -timekeeper : Label
-circleOne : Circle
-circleTwo : Circle
<<Property>> -alResult : Label
<<Property>> -leResult : Label
<<Property>> -abbResult : Label
-crt : CirclesThread
-tmk : Timekeeper
-c : Control
+AlgorithmsRaceGUI()
−runRace(event : ActionEvent) : void
+animateCircles(c1R : double, c2R : double) : void
+updateTime(time : String) : void
+startCircleAnim() : void
+startTimekeeper() : void
-arGUI
-algorithmsRaceGUI
Main
-algorithmsRaceGUI : AlgorithmsRaceGUI
+Main()
+main(args : String[]) : void
+start(primaryStage : Stage) : void
algorithms-race.fxml
model
Control
-gen : Random
-gui : AlgorithmsRaceGUI
-al : AL
-abb : ABBHandler
-le : LEHandler
-alRace : RaceThread
-abbRace : RaceThread
-leRace : RaceThread
+Control(alGUI : AlgorithmsRaceGUI)
+startRace(N : long, mode : String, algorithm : String) : void
+raceRunning() : boolean
-c
-gui
-al
AL
-al : ArrayList<Long>
+AL()
+addIterative(n : long) : void
+searchIterative(n : long) : boolean
+removeIterative(n : long) : boolean
+addRecursive(n : long) : void
+searchRecursive(n : long) : boolean
-searchRecursive(n : long, idx : int) : boolean
+removeRecursive(n : long) : boolean
-removeRecursive(n : long, idx : int) : boolean
-dataStructure
LEHandler
-first : LE
+LEHandler()
+addIterative(n : long) : void
+searchIterative(n : long) : boolean
+removeIterative(n : long) : boolean
+addRecursive(n : long) : void
-addRecursive(aux : LE, newEle : LE) : void
+searchRecursive(n : long) : boolean
-searchRecursive(aux : LE, n : long) : boolean
+removeRecursive(n : long) : boolean
-removeRecursive(aux : LE, n : long) : boolean
-le
-first
LE
<<Property>> -n : long
<<Property>> -next : LE
+LE(number : long)
+equals(n : long) : boolean
-next
DataStructure
+DataStructure()
+addIterative(n : long) : void
+searchIterative(n : long) : boolean
+removeIterative(n : long) : boolean
+addRecursive(n : long) : void
+searchRecursive(n : long) : boolean
+removeRecursive(n : long) : boolean
-abb
ABBHandler
-root : ABB
+ABBHandler()
+addIterative(n : long) : void
+searchIterative(n : long) : boolean
+removeIterative(n : long) : boolean
-findSuccessor(node : ABB) : ABB
+addRecursive(n : long) : void
-addRecursive(nodeInTree : ABB, newNode : ABB) : void
+searchRecursive(n : long) : boolean
-searchRecursive(nodeInTree : ABB, n : long) : boolean
+removeRecursive(n : long) : boolean
-removeRecursive(node : ABB, n : long) : ABB
-minValue(node : ABB) : ABB
-root
ABB
<<Property>> -n : long
<<Property>> -parent : ABB
<<Property>> -leftNode : ABB
<<Property>> -rightNode : ABB
+ABB(n : long)
+compareTo(num : long) : long
+equals(value : long) : boolean
-parent
-rightNode
-leftNode