Algorithm Analyzer Prototype Technical Manual
Based on the Thesis: Posteriori Analysis of Algorithms
Through Derivations of Growth Rate Based on Frequency Counts

Proponent:
John Paul Guzman

Adviser:
Teresita Limoanco

CLASS	Experimenter
SUPERCLASS	
PROPERTIES	int endTerm --Specifies the last term to be generated from the experiments int algoFilePath --Input algorithm file path String augmentedAlgoFilePath --Augmented algorithm file path float[] seqFreqCount --Generated data from the experiments
METHODS	Experiementer() Decimal log(float x) --Allows the evaluation of function strings with "log" function Decimal exp(float x) --Allows the evaluation of function strings with "exp" function Decimal pow(float base, float exponent) --Allows the evaluation of function strings with "pow" function void inputParameters(String algoFilePath, endTerm int, precision int) --Updates the variables used in the experiments String makeSpace(int tabAmount) --Generates whitespaces for the augmentation process void augmentCode() --Adds frequency count incrementers for each line instruction void runAugmentedAlgorithm()

	--Starts experimentation and stores the data void runInputFunction(String functionString) --Creates data using an input mathematical function void runInputSequence(String sequenceString) --Creates data using an input sequence of floats

CLASS	DataProcessor
SUPERCLASS	
PROPERTIES	int toleranceRound --Number of decimal places at the end to be rounded off (manages the errors caused by inaccuracy of floating point values) float[] a --Data used in the calculations int startTerm -- Specifies the first term to be generated from the experiments int endTerm --Specifies the last term to be generated from the experiments float[] e --Values of the e approximations assuming the "log" case float[] p --Values of the p approximations assuming the "log" case float[] c --Values of the c approximations assuming the "log" case float[] enl --Values of the e approximations assuming the "no log" case float[] pnl --Values of the p approximations assuming the "no log" case float[] cnl --Values of the c approximations assuming the "no log" case int[] discontinuities

	--The values of x at which discontinuities occurs boolean aroundDiscontinuities --Decision whether to calculate around the discontinuities or neighboring points boolean hasLog --Decision whether to choose the "log" case or "no log" case String hasLogPercent --Percentage of "log" case votes String hasLogRatio -- Ratio of "log" to "no log" case votes boolean converges --Decision whether to consider the data and approximations as asymptotically equivalent String convergesPercent --Percentage of "converges" votes String convergesRatio ---- Ratio of "converges" to "diverges" case votes
METHODS	DataProcessor() Decimal logg(float x) --Returns the natural log of x in decimal precision Decimal expp(float x) --Returns the natural exponentiation of x in decimal precision void inputParameters(float[] experimentalData, int startTerm, int endTerm, float k, int precision, int tolerance) --Updates the variables used in the calculations float[] removeConstants(float[] experimentalData, float k) --Subtracts a constant from all the elements of experimentalData in such a way that k is the value of the element at index 0. float[3] chooseMinimalErrorTerm(int term, boolean assumeDiscontinuous) --Chooses x, y, z mentioned in Appendix B of the thesis document.

	void calculateAsymptoticApproximations(boolean assumeDiscontinuous) --Calculates the e, p, c approximations void compareDiscontContApprox() --Heuristic that decides "aroundDiscontinuities" void checkForDiscontinuities() --Heuristic that detects discontinuities and stores them to "discontinuities" void determineHasLog() --Heuristic that decides "hasLog" void verifyAsymptoticEquivalence() --Heuristic that decides "converges" void generateEPCSequence() --Generates the plot points of the e, p, c approximations float roundOff(Decimal decimalValue, int decimalPlaces) --Rounds off a value to a given decimal place.

CLASS	GUI
SUPERCLASS	
PROPERTIES	gtk.Builder guiElements[] --Shorthand for all the GUI elements, including the ones extracted externally from the GUI.glade file.
METHODS	GUI() void addListener(String widgetName, Event event, callback action) --Adds a listener to a GUI element/widget void show() void quit() --Closes the main window void setText(String textViewName, String[] messageList) --Updates the contents of a widget

	void createGraph(int[] plotFreqCount, float[] plotEPC) --Create and shows the graph of the experimental data together with the e, p, c approximations.

CLASS	Controller
SUPERCLASS	
PROPERTIES	GUI gui --Gives access to a GUI (view) Experimenter experimenter --Gives access to an Experimenter (model) DataProcessor dataProcessor --Gives acces to a DataProcessor (model)
METHODS	Controller() void compute(gtk.Button widget) --Resets all the variables and generates data depending on the button clicked (e.g. if the "Compute: Algorithm" button is clicked, it generates data from experiments) void runCalculations() --Calculates all the approximations and runs all the heuristics void displayToGUI() --Update the GUI elements to reflect the latest values of the variables void graph() --Shows the graph void showAdvancedSettings() --Shows the settings containing the abstracted parameters void hideAdvancedSettings() void viewInput() --Opens the chosen algorithm file in an external text editor void roundOffSummary(gtk.SpinButton decimalPlacesSpinButton) --Shows the summary of all the significant values from the calculations and rounds off the approximations depending the SpinButton value

UML Class Diagram