Data Dictionary - Genetic Algorithm by Bradley McInerney

Catagory	Field Name	Type	Comment
GUI section - No function	root	<class 'tkinter.Tk'>	Holds all the data for the GUI window
	characters	<class 'list'>	Holds characters that the input and processing an use
	choice_mutateParents	<class 'tkinter.IntVar'>	Saves input data after GUI closes
	choice_outBasic	<class 'tkinter.IntVar'>	Saves input data after GUI closes
	choice_outGoal	<class 'tkinter.IntVar'>	Saves input data after GUI closes
	choice_outSubjects	<class 'tkinter.IntVar'>	Saves input data after GUI closes
	choice_graph	<class 'tkinter.IntVar'>	Saves input data after GUI closes
	graphStyle	<class 'list'>	Holds graph types for matplotliband display names for the GUI display
	menuBar	<class 'tkinter.Menu'>	Holds information for the menubar
	menuAbout	<class 'tkinter.Menu'>	Holds information for the menubar
	menuHelp	<class 'tkinter.Menu'>	Holds information for the menubar
	entry_goal	<class 'tkinter.Entry'>	An entry box in the GUI (Needs a variable to get information from)
	entry_mutationChance	<class 'tkinter.Entry'>	An entry box in the GUI (Needs a variable to get information from)
	entry_popSize	<class 'tkinter.Entry'>	An entry box in the GUI (Needs a variable to get information from)
	entry_popTarget	<class 'tkinter.Entry'>	An entry box in the GUI (Needs a variable to get information from)
	entry_genLim	<class 'tkinter.Entry'>	An entry box in the GUI (Needs a variable to get information from)
	check_mutateParents	<class 'tkinter.Checkbutton'>	A check box in the GUI (Needs a variable to get information from)
	graph_preset	<class 'tkinter.Button'>	A button in the GUI (This particular button needs a variable so it can be disabled)
	check_outBasic	<class 'tkinter.Checkbutton'>	A check box in the GUI (Needs a variable to get information from)
	check_outGoal	<class 'tkinter.Checkbutton'>	A check box in the GUI (Needs a variable to get information from)
	check_outSubjects	<class 'tkinter.Checkbutton'>	A check box in the GUI (Needs a variable to get information from)
	graphSelection	<class 'list'>	Holds information for radio buttons in the GUI
	runConfirm	<class 'bool'>	Checks if the program was run or not after the window has closed
GUI section - showTextInfo()	types	<class 'dict'>	Holds the text to be presented inside every text popup
GUI section - act_run()	returned_goal	<class 'str'>	Stores data for use after rthe GUI closes
	returned_mutationChance	<class 'str'>	Stores data for use after rthe GUI closes
	returned_popSize	<class 'str'>	Stores data for use after rthe GUI closes
	returned_popTarget	<class 'str'>	Stores data for use after rthe GUI closes
	returned_genLim	<class 'str'>	Stores data for use after rthe GUI closes
	returned_mutateParents	<class 'int'>	Stores data for use after rthe GUI closes
	returned_outBasic	<class 'int'>	Stores data for use after rthe GUI closes
	returned_outGoal	<class 'int'>	Stores data for use after rthe GUI closes
	returned_outSubjects	<class 'int'>	Stores data for use after rthe GUI closes
	returned_graph	<class 'int'>	Stores data for use after rthe GUI closes
	runError	<class 'list'>	Checks for any invalidations that would cause errors
	missing	<class 'list'>	Checks if characters used are missing the the valid characters list
After GUI closes - No function	goal	<class 'str'>	Gets data from returned_goal after the GUI closes
	mutationChance	<class 'int'>	Gets data from returned_mutationChance after the GUI closes
	maxPop	<class 'int'>	Gets data from returned_popSize after the GUI closes
	targetPop	<class 'int'>	Gets data from returned_popTarget after the GUI closes
	maxGen	<class 'int'>	Gets data from returned_genLim after the GUI closes
	onlyMutateChildren	<class 'bool'>	Gets data from returned_mutateParents after the GUI closes (Opposite of value)
	outputGenStats	<class 'bool'>	Gets data from returned_goal after the GUI closes
	outputGenGoal	<class 'bool'>	Gets data from returned_goal after the GUI closes
	outputSubjects	<class 'bool'>	Gets data from returned_goal after the GUI closes
	graph	<class 'bool'>	Gets data from returned_goal after the GUI closes
	graphStyle	<class 'str'>	The style (previously stored in graphStyle) the matplotlib reads as a graph style
	population	<class 'list'>	Hold all the strings that are subjects to be compared to the goal
	generation	<class 'list'>	Holds all populations created
	statHighest	<class 'list'>	Holds the highest score that was achived in a population
	statLowest	<class 'list'>	Holds the lowest score that was achived in a population
	statMedian	<class 'list'>	Holds the median score that was achived in a population
	Average	<class 'list'>	Holds the average score that was achived in a population
	population2	<class 'list'>	Between stages of processing, this holds the data of population
	fitness	<class 'int'>	The score for the current subject being processed
	total	<class 'bool'>	Hold the scores in a population added together. Used to find the average
	correctSubjects	<class 'int'>	Holds the number of perfect subjects to be compared with the tagetPop
	loop	<class 'int'>	Used to hold position while the program kills off the bottom 90% of subjects from the population
	child	<class 'str'>	Holds the data of a child subject while it is "bred" from the parents
	newSubject	<class 'str'>	Holds the data of a subject while it is mutated
	output	<class 'list'>	Holds the closest subject to the goal if none are perfect
	highest	<class 'list'>	Holds the value of the highest score achieved
	x	<class 'list'>	Hold the x values for the graph