Conic Plotter Script Version 1.1 Quick Instruction Listing

Andy Cox V

Introduction:

This programming language was developed for an Algebra II project which the goal was to draw an image using conics. For version 1 I started the project on March 2, 2017 and ended work on it on March 27, 2017. Note the work for March 28, 2017 was not recorded and consisted of miscellaneous notes not related to the program such as an instruction manual. On version 1 I spent a total of 19 days and on version 1.1 I spent one day April 14, 2017. For the total project I spent 20 days. Both the programs have not been compiled as of April 14, 2017. The programs run in the IDE properly and are programmed in Java. Due to the due date March 28, 2017 I was not able to test all of the attributes but to my knowledge all of the instructions unless stated otherwise should work as intended. Version 1 was the program used to plot the conic image that was graded for the Algebra II project on March 28, 2017. On June 17, 2018 this file was corrected for grammer and formating errors.

Program Source Files Totals				
File Name	**Lines**	**Words**	**File Length in Bytes**	**Characters without white space**
ConicConstant.java	108	608	6309	6092
ConicData.java	332	1400	12692	12030
ConicPlotter.java	211	891	7446	7026
ErrorHandling.java	121	378	3442	3202
Interpreter.java	176	843	8991	8641
PlotPoints.java	415	1375	10836	10008
PrintAttributes.java	413	1541	15337	14513
TOTALS	***1776***	***7036***	***65053***	***61512***

Argument Notes:

Throught this guide arguments for instructions will have succeeding letter keys to represent their argument type(s). These keys are represented below:

- All instructions are case sensitive/all lowercase.
- x and y may be either an integer or a variable.
- z must be a variable.
- Acc is the accumulator a register which can not be directly accessed by the user used for program operators.
- := equals assign to where in x := y, x is assigned to the value of y.

Arithmetic Operators

All arithmetic operators change the value of the accumulator based on the argument(s) given.

add x y - Acc := x + y
subtract x y - Acc := x – y
divide x y - Acc := x / y
multiply x y - Acc := x * y
squareroot x - Acc := √x
square x - Acc := x^2
negate x - Acc := -x

Variable Manipulation

giving z - z := Acc
set x y - x := y
swap - x := y and y := x

Console Interface

read z - z := Integer Input From User
startdisplayblock - Prints all strings as literals unless $displayspecial: is called then the following argument is displayed.
enddisplayblock - Terminates startdisplayblock.
$displayspecial: - Can only be executed when the startdisplayblock instruction is active.
Arguments succeeding **$displayspecial** are:
space - Displays a single space character ‘ ‘.
newline - Displays a new line Java escape “\n”.
carrigereturn - Displays a carriage return Java escape “\r”.
tabulation - Displays a tab Java escape “\t”.
formfeed - Displays a form feed Java escape “\f”.
$displayspecial: - Literal display of string “*$displayspecial:”.*
x - Where x is a variable displays the value of the variable.

Branching

goto x - Searches for a label with the same value as x and executes succeeding instructions.
label x - Used as a location for goto to branch to.
compare x y - Sets an internal flag in the program used for conditional branching.
goifgreaterthan x - If the most current compare instruction sets the

internal flag as true if x > y then go to label x.

goiflessthan x - If the most current compare instruction sets the internal flag as true if x < y then go to label x.

goifequalto x - If the most current compare instruction sets the internal flag as true if x = y then go to label x.

goifnotequalto x - If the most current compare instruction sets the internal flag as true if x != y then go to label x. If a compare instruction has not been executed yet this instruction will execute when reached.

Conic Plot Functions

point x y - Load values from variable registers a and b to plot a single point.

linesegment x y x2 y2 - Load values from variable registers a, b, c, and d. Where (x,y) is one point and (x2,y2) is another point to plot a line segment.

verticalparabola x x x x - Load values from variable registers a, b, c, and d. Where a and b are the x and y shifts, c is the focal point, and e is the range.

horizontalparabola x x x x - Load values from variable registers a, b, c, and d. Where a and b are the x and y shifts, c is the focal point, and e is the domain.

circle x x x - Load values from variable registers a, b, and e. Where a and b are the x and y shifts and c is the radius.

ellipse x x x x - Load values from variable registers a, b, c, d. Where a and b are the x and y shifts, c is the a and d is the b in the standard equation.

verticalhyperbola x x x x - Load values from variable registers a, b, c, d, e. Where a and b are the x and y shifts, c is the hyperbola a value, d is the hyperbola b value, and e is the range.

horizontalhyperbola x x x x - Load values from variable registers a, b, c, d, e. Where a and b are the x and y shifts, c is the hyperbola a value, d is the hyperbola b value, and e is the domain.

Commenting

comment - A single line comment. Comments all code until the next new line character(s) have been reached.

startblockcomment - Starts a block comment all preceding text until endblockcomment is reached will be commented out.

endblockcomment - Terminator for the startblockcomment instruction

Variable Listing

This is a listing of all variables which are signed integers. Note there are 26 variables all uppercase letters A to Z.

A B C D E F G H I J K L M N O

P Q R S T U V W X Y Z

Conic Plot Register Variables

These variable names access and alter the internal registers used by the conic plot functions. There are five internal variables named a, b, c, d, and e. This is a list of the strings that are used to represent manipulation with the associated register. These are treated as variables by the script program.

A - #xset, #xshift (Used for x axis shifting)

B - #yset, #yshift (Used for y axis shifting)

C - #simitransverseaxis, #simimajoraxis, #focaldistance, #xtwoset, #a, #p (Used by various conics)

D - #simiconjugateaxis, #simiminoraxis, #ytwoset, #b (Used by various conics)

E - #domain, #range, #radius (Used as the domains and ranges of conics and radius for the circle conic function)

Program Termination

If the program reaches an End of File character the script will terminate or if the exit instruction is initiated.

exit - Terminates the script.

Bug Listing

- Both horizontal and vertical Hyperbolas and Parabolas do not plot out their respected ranges and domains. Instead the values that are set to be the domain and range are used to plot out how many pixels on each side the conic should be composed of. For example if a horizontal parabola was set to have a domain of seven it will only plot out seven pixels on each side.
- Parabolas plot out inverted focal distances. For example a parabola with a positive focal distance will plot down and a parabola with a negative focal distance will plot up. Which is reverse of how the parabolas should be plotted.
- Ellipses should have better definition for the simi-major and simi-minor axis. Basically there should be a definite horizontalellipse instruction and a verticalellipse instruction to plot both vertical and horizontal ellipses.
- On rare occasion blank space will not be cropped out by the image.