

Version 1.0	Tasks:	Notes on Task:	Start Date:	Initiated By:	End Date:	Finished By:	Notes:
	Meeting 1: Planning 7/5	First meeting to go over project specifications, our goals and preliminary ideas to start the project, as well as construction of major milestones and a breakdown of the first milestone. As well as setting up our member contribution sheet and our repository.	7/5	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Absent - Sick)	7/5	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Absent - Sick)	Thoroughly discussed the project as a whole. Split the initial algorithms into three sections. Aeyva - Curve. Rebecca and Alexandria - Triangle. Nathaniel - Snowflake. With the intentions of combining our programs into a simple conditional based user interface that will ask the user which algorithm they'd like to run, followed by a request for necessary input, and then calling the respective algorithm to output an image of the recursive graphic. Eventually , upon completion of the 3 assigned graphics we will work together as a group to create a fourth graphic of some kind. Discussed the basic gameplan for our report/member contribution sheet / and how to properly use our repository. Set next meeting date - with the goals of getting barebones of our respective assignments together. As well as generic construction of a user interface that we will insert our individual code into after submission finalization.
	Meeting 2: Work 7/11	Report back on work with our barebones code. Figure out where to move forward, and set new tasks required for finalization of version 2. *Version 1 will be completed by this day.	7/11	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Present)	7/11	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Present)	General barebone and pseudocode has been created for each of the algorithms. Decided the best way to move forward from here is to continue implementing our programs. Koch's Snowflake needs the recursion element added. Traingle needs base case draw function and recursive elements. Curve needs base case draw function and recursive elements. Similarly we need to work as a group to figure out a save image functionality using our SFML library. As well as introducing a parameter menu to customize our graphics. Thinking even further; into V.3 development; we have decided to meet every day next week after finalization of Version 2.0 in order to write the Report and finalize documents needed for submission. As well as practicing our project presentation with our last day together.
	Objective 1: Curve Program	Start the barebones of the curve program; research algorithm and start construction of the .cpp functions/ classes related to the topic.	7/5	Aeyva	7/11	Aeyva	Completed the installation of the SFML library, as well as the pseudocode for the Hilbert's Curve algorithm. Detailed the plans for a recursive call feature to properly iterate through the curves algorithm. Started implementation of initial class files and construction of the draw base case function. Needs to be finished; then working on implementing the recursive call.
	Objective 2: Snowflake Program	Start the barebones of the snowflake program; research algorithm and start construction of the .cpp functions/ classes related to the topic.	7/5	Nathaniel Brown	7/10	Nathaniel Brown	Finished; Slightly more than the skeleton. Deep dived on the template classes within the SFML library. Started following the algorithm for the Koch's Snowflake and managed to get a functioning class together that can output triangles. Not entirely finished; needs methods and recursive element to properly implement the snowflake recursion call; but the base is there.... Also needs reoptimization to allow for the collection of user input to determine construction aesthetics. i.e. ask for color, stage of the snowflake, filename. As well as implementation of save as png function. TL;DR:V1.0 is completed. Program can construct a triangle. Developed helpers that should be usable for recursive calls as well! New goals will be set after meeting 2 on 7/11
	Objective 3: Triangle Program	Create project - add pseudocode for planning of program - begin development	7/5	Rebecca & Alexandria	7/8	Rebecca	Completed set up of SFML library. Got template project working. Completed pseudocode to implement coding plan.
	Objective 4: Install & Configure SFML	As a group; figure out how to properly configure SFML so we can actually start programming. (This is **strictly** an objective as adding a library has not been covered in any course material at URI thusfar)	7/5	All Members	7/8	Nathaniel Brown	This was a pain; but proper installation methods have been heavily detailed and shared in this document. This document will need slight editing and formatting for input within our final project report.

	Task 1: Basic User Interface	Throw together a generic conditoinal based user interface; which will be used to call each of the individual algorithms the group works on.	7/5	Nathaniel Brown	7/5	Nathaniel Brown	Finished. Basic template has been constructed. Consisting of a main function that will be used to call various helper void display function. The program prompts the user for a main menu user input; then directs the user to the appropriate algorithms display message and prompts for whatever parameters are required for that algorithms output. This will be adjusted as Objectives 1 through 3 are finished. Invalid data outputs and redirects for inputs out of range within the main menu. Additional validity checks can be addressed in the future for parameter constrictions.
Version 2.0	Tasks:	Notes on Task:	Start Date:	Initiated By:	End Date:	Finished By:	Notes:
	Meeting 3: Update and Work 7/13	Meet with updated code for the three algorithms; see where we are at and establish more accurate tasks as needed.	7/13	All Members	7/13	All Members	Met with mostly finalized code for the three algorithms; evaluated what needed to be done to finish implementation for the project and set tasks that needed to still be handled.
	Meeting 4: Update and Work 7/18	Meet with finished implementations of algorithms; and combine everything into a master file for submission.	7/18	All Members	7/18	All Members	Met with mostly finalized code for the three algorithms; evaluated what needed to be done to finish implementation for the project and set tasks that needed to still be handled.
	Objective 1: Implement Koch's Snowflake Recursion Aspect	Work on a recursive helper to continuously call the triangle code made in Version 1.0	7/13	Nathaniel Brown	7/15	Nathaniel Brown	Recursive function is completed. Program now works for various stages of Koch's Snowflake algorithm; however it does crash on stages higher than 10. This is likely due to hardware; not code. I will begin additional enhancements to the program to include some level of customizability via parameters in the user interface.
	Objective 2: Implement Kierpinski's Triangle	Implement Triangle algorithm utilizing the SFML and handle proper creation of related objects and construction of the base case drawing.	7/13	Rebecca & Alexandria	7/16	Rebecca & Alexandria	Created the initial Triangle creation algorithm and worked partway through the recursive elements. More implementation and work needs to be done to complete the recursive call as intended.
	Objective 3: Implement Hilbert's Curve	Implement curve algorithm utilizing the SFML and handle proper creation of related objects and construction of the base case drawing.	7/13	Aeyva	7/15	Aeyva	Handled the construction of the initial base case graphic. Need to continue the recursive elements and iterations of the program as a whole.
	Task 1: Save as Image Function	Implement a function that can be called to save the graphic of our algorithms to an image. As per project requirements.	7/13	All Members	7/15	Nathaniel Brown	A function was not possible due to the nature of the template class Render Window; could not pass a current state of the rendered graphic into another function to save. Therefore the save image functionality needs to be manually written for each version of our windows. The method has been created and distributed amongst the group.
	Task 2: Configure Submenu Parameters	Implement some level of customizability of your graphic via setting parameters within the user interface.	7/13	All Members	7/19	All Members	Needed to wait until finalized code was in for all members; submenu has been altered to recieve the appropriate parameters needed for output. I.e. File names, color, and stage where necessary.
Version 3.0	Tasks:	Notes on Task:	Start Date:	Initiated By:	End Date:	Finished By:	Notes:
Final Version of code. Anything other than touch-up work was outside of expected finish time.	Meeting 5: 7/19	Meet to finalize program and piece separate programs together; troubleshoot any remaining issues. Get submission in by the end of the 19th.	7/19	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Absent - Family Emergency)	7/19	Nathaniel (Present), Rebecca (Present), Aeyva (Present), Alexandria (Absent - Family Emergency)	Programs are not completed for the CURVE or the TRIANGLE; those projects need to be hyper focused over night and brought to a conclusion. So we can start presentation practice for tomorrow. Report is 80% done missing instructions for the programs that are not fully implemented.
	Meeting 6: 7/20	Practice presentation; finalize the remaining bits of documents for submission.	7/20	All Members	7/20	All Members	Finalized documents; submitted and practiced presentation for Tuesday at 3:30pm.

	Curve Objective 1: Outline Data Structure, Methods, and Variables	Develop getters and setters necessary for the algorithms' needs, as well as outline base methods required such as 'drawCup' and 'drawLine'	7/18	Aeyva & Alexandria	7/19	Aeyva & Alexandria	Curve's implementation is outlined with the basic functions, member variables within the curve object, and the overall call stack to simulate Hilbert's Curve.
	Curve Objective 2: Calculation Solutions	Need to solve for the equations to implement a scalable implementation of the curve	7/19	Aeyva & Alexandria	7/20	Aeyva & Alexandria	Design a scaled size variable that will help subdivide the small square chunks properly and offsets appropriately, as well as other necessary computations for the curve methods, numerical formulas for number of cups and curves based on the order of the curve, etc.
	Curve Objective 3: Demonstrate Algorithm Design	Algorithm from start to finish needs to be demonstrated for group understanding and to push out the final stages of the curve collaboratively.	7/19	Aeyva & Alexandria	7/20	Aeyva & Alexandria	Curve algorithm is the last implementation needed, collaborate with team members for a full understanding on a working solution to it, starting from the input, to the member variables needed to prime the recursive algorithm, consistently implement the higher order cases based on the first order case chosen.
	Curve Objective 4: Recursive Functions	Develop recursive call to implement various stages of algorithms graphic.	7/19	Aeyva & Alexandria	7/22	Aeyva & Alexandria	Recursive function was solved and implemented by Alexandria; needed further calculation adjustments to display appropriate graphic.
	Curve Object 5: Fix Point Calculations/Calls	The way points are passed needs to be reformatted; to display the appropriate graphic.	7/22	Aeyva	7/22	Aeyva	Solved; graphic is now appropriately displaying.
	Curve Rescue Objective:	Code was not implemented by the 7/18 due date. During last meeting Alexandria was starting to assist Aeyva in development of the final algorithms code. They broke off and both worked on separate versions to try and get the program started.	7/20	Alexandria	7/22	Alexandria	Since Alexandria couldn't contribute significantly to the Triangle, and Aeyva did not have their Curve working or implemented past theory on 7/18 . Alexandria started to breakdown the Curve algorithm and implement her own code. She got the code working over the weekend. On 7/21 she had a fully functioning graphic; albeit with a few off lines. From here Aeyva was able to complete the curve.
	Triangle Objective 1: Develop Helper Function	Create function that outputs a midpoint between two given coordinates	7/19	Rebecca Iselin	7/19	Rebecca Iselin	Developed necessary helper functions that can be used within the initiate sierpinski function. They help calculate midpoints and establish coordinates needed for a future recursive call function
	Triangle Objective 2: Develop Recursive Function	Create function that recursively draw a triangle given 3 midpoints, then iterates and calls itself on newly defined midpoints.	7/19	Rebecca Iselin	7/19	Rebecca Iselin	Developed the recursive function which uses coordinates established via helpers; to decide where to draw recursive triangles on the graphic.
	Triangle Objective 3: Finalize	Implement into main; tweak parameters.	7/19	Rebecca Iselin	7/19	Rebecca Iselin	Finalized parameters; tweaked calculations slightly to make the triangle more clean. Revamped some of the main menu displays; cleaned up the main file. And implemented the curve.
	Snowflake Objectives: Implement STAGE parameter	No snowflake objectives are needed at this point; algorithm was completed on schedule. Additional parameter is being added to allow the program to output a graphic at a particular iteration given by the user.	7/19	Nathaniel Brown	7/19	Nathaniel Brown	Parameter implemented; now the user can set the stage, rather than the stage of iterations being predetermined by the hard code within the snowflake.cpp
	Bonus Objective: File Saving Location	Edit pathing for file save.	7/18	Rebecca Iselin	6/18	Rebecca Iselin	Updated the file name directory saving; now it will output our png images into the working directory; NOT within the testing folder within C-make
	Bonus Objective: Modularized Color Parameter	Create a helper function for the ui; to reduce rewritten code	7/19	Rebecca Iselin	7/19	Rebecca Iselin	getColor function has been created for the the parameter requests within the basic UI for our program.
REPORT	Tasks:	Notes on Task:	Start Date:	Initiated By:	End Date:	Finished By:	Notes:
	REPORT - Task 1: Introduction Project	Write an introduction to the project as a whole; how we approached it, all algorithms used.	7/19	Nathaniel Brown	7/19	Nathaniel Brown	Introduction completed.
	REPORT - Task 2: Introduction Topic	Write an introduction to the topic; what are recursive graphics, how are they used, why are they used.	7/19	Nathaniel Brown	7/19	Nathaniel Brown	Introduction completed.
	REPORT - Task 3: Instructions Compile	Write detailed instructions to compile and run our program.	7/19	Rebecca Iselin	7/22	Rebecca Iselin	Had to wait for late work to be turned in; finished the compilation instructions for the final section of our overall code.

	REPORT - Task 4: Instructoins SFML	Write detailed instructions on SFML installation and setup.	7/19	Nathaniel Brown	7/19	Nathaniel Brown & Alexandria	Detailed instructions have been copied over and reformatted from the PDF I created early on in the project explaining how to set up SFML. Nathaniel handled PC, Alexandria handled MAC
	REPORT - Task 4: SFML Info	Write sectoin of body explaining SFML template classes.	7/19	Nathaniel Brown	7/19	Nathaniel Brown	Section completed; details various class templates and some of their helper functions and methods.
	REPORT - Task 5: Triangle Breakdown	Write section of body explaining Algorithm in depth.	7/19	Alexandria	7/20	Alexandria	Algorithm for hte tirangle described and visualizations were created.
	REPORT - Task 6: Curve Breakdown	Write section of body explaining Curve algorithm in depth	7/19	Aeyva Rebelo	7/20	Aeyva Rebelo	Algorithm is sufficiently explained, detailed the construction of the points based on the quadrants, the general shape of the curve algorithm, and how to go about constructing higher orders using images
	REPORT - Task 7: Snowflake Breakdown	Write section of body explain snowflake algorithm in depth.	7/19	Nathaniel Brown	7/19	Nathaniel Brown	Algorithm is explained; included a section detailing various stages of koch's snowflake as well; to emphasize the importance of the iterative recursive calls.
	REPORT - Task 8: Conclusions	Together; write a conclusion section relating the algorithms and importance of recursion. Include possible scenarios in which recursive graphics are used in the real world.	7/19	Aeyva	7/22	Aeyva	Conclusion completed. Real world possible applications included.
	REPORT - Task 9: Formatting and ToC	Take the google document and reformat within word; for submission.	7/19	Rebecca Iselin	7/20	Rebecca Iselin	Had to wait for late work to be turned in; finished the formatting of the entire document using the shared work in the google.doc
PRESENTATION	Tasks:	Notes on Task:	Start Date:	Initiated By:	End Date:	Finished By:	Notes:
	PRESENTATION - Slideshow and Practice	As a group, create presentation slideshow and practice presenting several times before Tuesday's presentation.	7/20	All Members	7/20	All Members	Algorithm Breakdowns: Nathaniel - Snowflake, Rebecca - Triangle, Aeyva - Curve. Introduction to topic, and introduction to project (Alexandria). Conclusion was written by Aeyva. SFML Introduction (Nathaniel)