Geometric Canvas

Ravi Tomar
School of Computer Science & Engineering, University of Petroleum and Energy Studies, Dehradun, India

Akshat Gupta
School of Computer Science & Engineering, University of Petroleum and Energy Studies, Dehradun, India

Ayush Patni
School of Computer Science & Engineering, University of Petroleum and Energy Studies, Dehradun, India

S. Muthazhagu
School of Computer Science & Engineering, University of Petroleum and Energy Studies, Dehradun, India

ABSTRACT
Geometry is one of the basic concepts and forms the main element of mathematics. So to make it simple and easy to understand Logo Programming language was designed with some geometric Functions. In the proposed work going to implement the functions and concept of logo Programming language through C Programming Language and C Graphics with Some additional features. The logo programming includes the turtle as its cursor which could be moved or directed anywhere by giving instruction to the computer. Now, the turtle was used as a cursor on computer graphics screen where it is used to draw shapes, designs and pictures. Hence, helping the students to understand line by line geometry. Logo programming may provide opportunities for improvement of creativity and problem solving skill.

Keywords
Programming, Turtle, C Graphics, Geometry

1. INTRODUCTION
Computational geometry is the study of algorithms that can be states in terms of geometry. Computational Geometry emerged in late 1970. Computational Geometry finds its application in almost every sector of today's world. The main areas of applications for Computational geometry are Robotics, Geographic information system, IC designing, Computer-aided engineering, computer vision. The main force behind the development of Computation geometry was the progress in computational graphics and computer-aided design and manufacturing. The branches that deal with computational geometry are Combinatorial Computational Geometry and Numerical Computational Geometry [1]. The computational geometry gave a solution to understand the problems of the geometric properties and also in proper application of algorithmic techniques (paradigms). Computational geometry aided to the enhancement for the study of algorithms for geometric objects by focusing more on exact algorithms that are asymptotically fast. The ultimate purpose of the research work is to represent geometry by providing wide options for the user to understand geometrical shapes and figures and the entire concept with the use of graphics in the field of primary and secondary education .This work is implemented through the use of basic C programming language and graphics files. The work enables the user to visualize the concepts in a more better way of understanding. The project drags the scope of the work in developing its own set of commands for the ease of user input by defining its own PENCIL MOVING ALGORITHM and making it more users friendly.

1.1 Previous Work
Logo was developed in 1967 by Daniel G. Bobrow, Walley feurzeig, Seymour Papert and Cynthia Solomon. Logo was derived from the Greek word "LOGOS" meaning "THOUGHT". It was developed to give a new way to programming that was basically numbers not logic oriented[2].

Logo is generally implemented as interpreted language. The interactive approach of log aids the programmer by providing immediate feedback on instructions and helps in debugging it easily. Programs in logo are collection of small procedures. Each procedure is written in text editor. The word **"to** "is used at the beginning of each procedure. Similarly to indicate the end of the procedure, **"end"** keyword is used. Until and unless you know the logo implementation, you cannot differentiate that the words are primitive or user-defined [3].

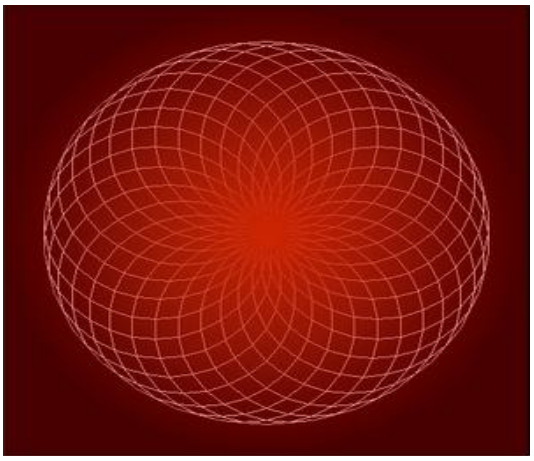

Fig1: A sample output of Logo Programming

Despite being a general-purpose language, Logo is often is known for its use of turtle graphics, in which the commands for movement or drawing produce output *i.e*., Line Graphics on screen, with or without robot which looks like turtle. The basic idea to develop "LOGO" was to teach students, the movements of turtle. Logo is based on the programming concept of "LISP". UCBLOGO can be used to teach all computer science concepts like, handling lists, files, I/O, and recursion in scripts.

1.2 Computer Graphics
The visual representation of objects/ figures on to a Display screen such as a monitor is one of the key areas that computer graphics deals with. This Field involves manipulation/modification of real world objects using appropriate transformation in order to display them on a Visual Display Unit (VDU). Algorithms have been

developed for improving the effects of picture generation [4].

Today's generation in Computer Graphics has been transformed from 2D to 3D. The integration of physics with the algorithms built for graphical processing is necessary in order to simulate real world effects. The Graphics Kernel System (GKS) was the first graphics package created for use through the combined efforts of ANSI and ISO. Now computer graphics has been developed totally in a new level. Where basic 2D animations are now converted into 3D. And Computer Generated Effects are the current trend of Computer Graphics [5].

2. DESIGN

Design refers to the representation of an entity incorporated with the components needed. It is an essential phase of testing software's quality since it provides a visual representation of the final product. Users can then inspect the design in order to determine whether the requirements are met or not. Firstly designed a separate function in project that will take process the input then made all set of functions that will perform the input and process the respective geometrical shapes. Then at the end implemented file handling to store the commands.

2.1 Flow Chart

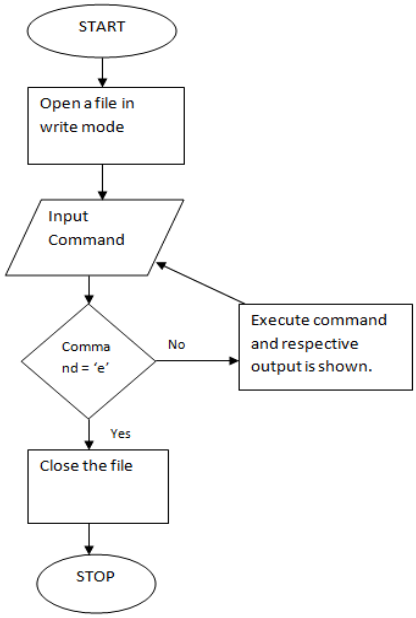

Fig 2: Flow Chart

2.2 0-Level DFD

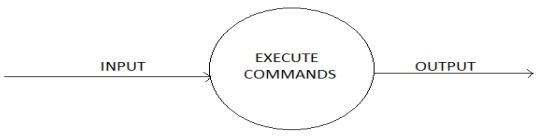

Fig 3: Level – 0 DFD

2.3 1-Level DFD

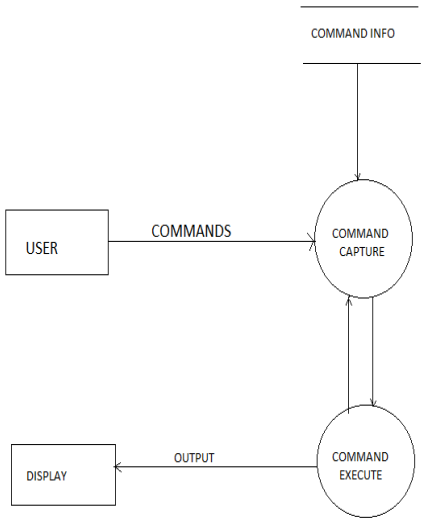

Fig4: Level – 1 DFD

2.4 Algorithm

1.x←x coordinate
2. y←y coordinate
3. x1← 2nd x coordinate
4. y1← 2nd y coordinate
5. len← To keep track of length/radius.
6. fp← file pointer
7. a[5]← array of 5 indexes to input direction & length from user.
8. dir← to keep track of direction.
9. gd← graphic driver.

BEGIN:

10. x←200
11. y←200
12.gd←DETECT
13. Setting path for graphic library.
14. Open file Canvas3.txt in "w" mode.
15. do-while(a[0]!='e')
16. Input &a from user.
17. Write the user value into the file.
18. dir←a[0]
19.len←((int)a[1]-48)*10+((int)a[2]-48);
20. Pass dir & len in function process(dir,len).
21. End while.
22. Close the file
23. *END*

//Start of function process(char dir, int len)

BEGIN:

24. k← len
25. if(dir is 'r')
26. Draw line towards right of length k;
27. Else if(dir is 'd')
28. Draw line vertically downwards of length k;
29. Else if(dir is 'l')
30. Draw line at left of length k;
31. Else if(dir is 'u')
32. Draw line upwards of length k;
33. Else if(dir is 'c')
34. Draw a circle with radius k;
35. Else if(dir is 'o')
36. Bring the pencil at initial position x←200,y←200;

Else if(dir is 't')

37. Draw equilateral triangle of length k;
38. Else if(dir is 'E')
39. Enter X and Y Radii and an draw ellipse with respective radius.;
40. Else if(dir is 'g')
41. Enter X and Y coordinates and move the pencil to given coordinates ;
42. Else if(dir is 'a')
43. Enter angle and draw the arc with radius k;
44. Else if(dir is 'C')
45. Enter value from 0-15 change the colour of the pencil.
46. Else if(dir is 'p')
47.Previous commands read from the file and executed.
48. End if.
49. END

2.5 Algorithm Explanation

The algorithm used in the project is PENCIL MOVING ALGORITHM. In this algorithm 200,200 an initial coordinates has been taken and the commands are given the track of previous co-ordinates are taken and the next instruction's output is proceeded from the last co-ordinates. graphics.h header file is used which consist of line(), circle, arc() functions. A file pointer is maintained which keeps the track of all the commands and saves in a file. Now the among 13 commands, a command 'p' which repeats the previous specified commands so if 'p' is input given by the user then file the current file is closed in write mode and reopened in read mode and all the previous commands are read from the file and operations are performed. Then to further add input other than 'p' the the file is opened in append mode then further commands can be input and can be recorded in file. All inputs are scanned from the user in form of an array and are passed in a function 'process(char dir, int len)' before passing to the function value at array[0] index is passed as a character while value at array[1] & array[2] are converted into integer and the passed to 'len'. Then in the process() there are 12 if-else conditions and according to the char value in 'dir' those if-else conditions are operated. In this line(), circle(), arc(),ellipse() and setcolor() functions are being used. The values get from 'process (char dir, int len)' are passed in these function and the operations are performed accordingly.

3. IMPLEMENTATION

The project has used the following module:
Iterative Waterfall Model

3.1 Methodology

Water Fall Software Development Process:
Analysis: The Analysis of requirements of the system and proper working windows operating system for the coding.
Design: In this study, using PENCIL MOVING Algorithm. The lines will move according to the commands that user will input.
Coding: Basically using C Graphics, Data Structure, and File Handling.
Testing: For different versions of Microsoft windows testing should be done and respective debugging should be implemented.
Documentation: After completion of the project Documentation is done by entire team.

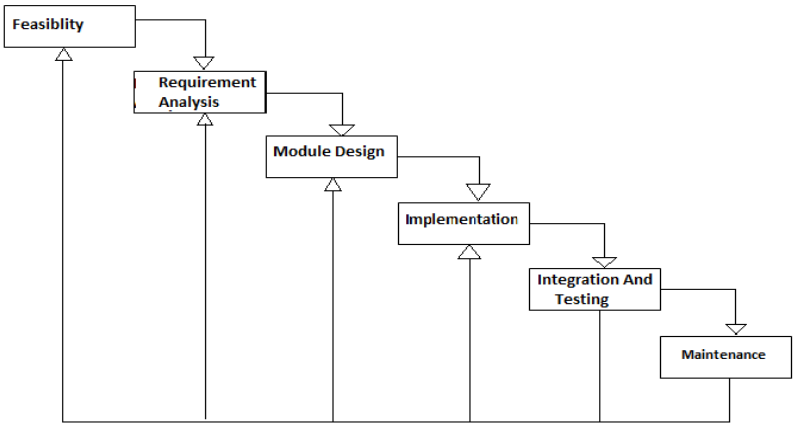

Fig 5: Stages of Iterative Waterfall Model

3.2 Output

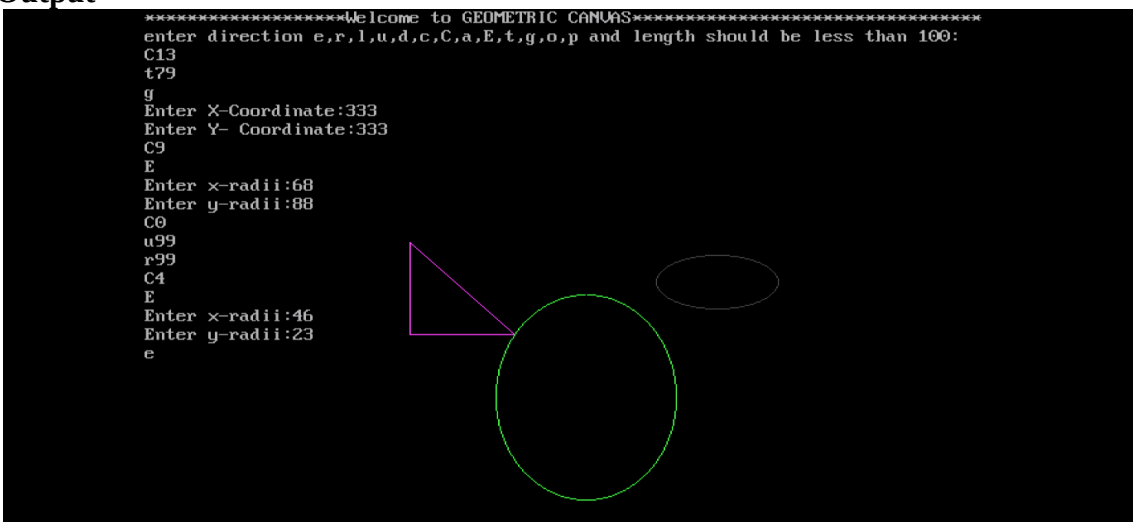

Fig 6: Output of basic geometric figure

In Figure 6, start to make an equilateral triangle simply use 't79' to do it. Then moving to another coordinates using 'g' to make a vertical ellipse type 'E' and give x-radii greater than y-radii. Then moving on to another coordinate using 'g', type 'E' and vice-versa to make a Horizontal Ellipse the exit.

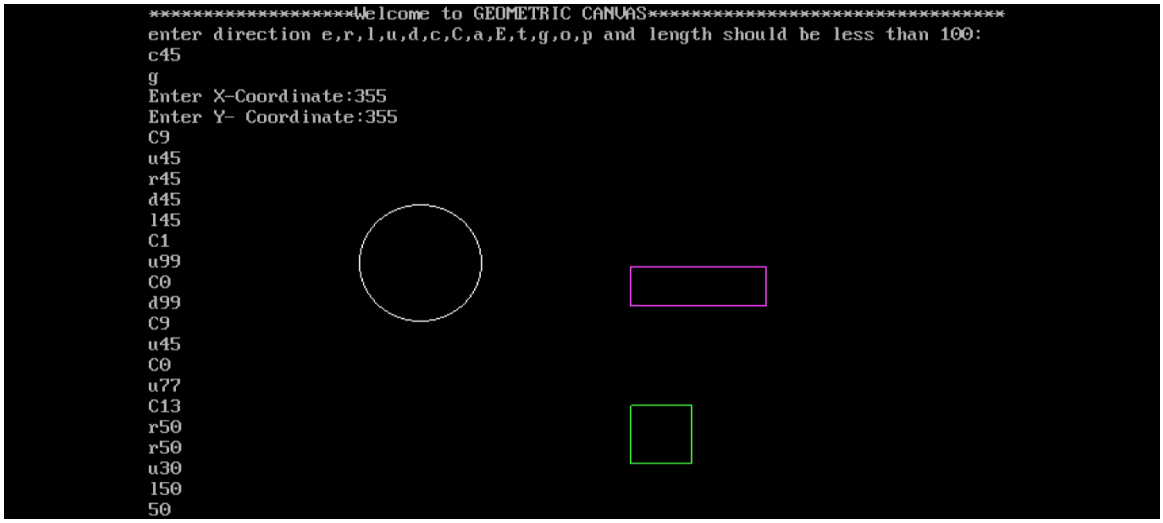

Fig 7: Output of basic geometric figure

In Figure7 to make a circle simply entered 'c45' and a circle with radius 45 is drawn. Move to another coordinate using 'g' and changed the color to purple using 'C9' command and drawn a rectangle with set of commands using u,l,r,d. Similarly moving to next coordinates and made a square using u, l, r,d commands and exited 'e'.

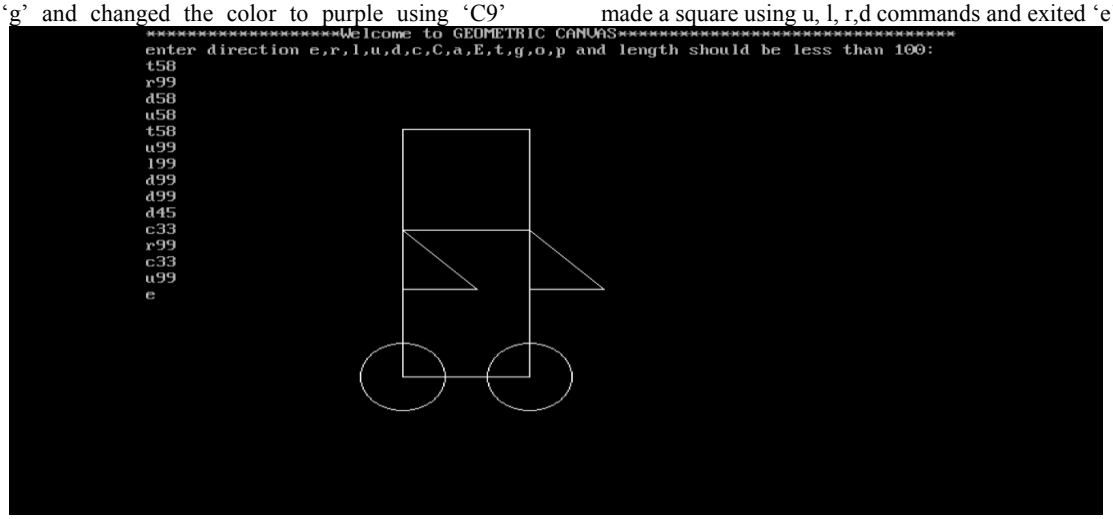

Fig 8: Output of combinations of basic geometric figures

Doing something creative in figure 8, try to make a 'toy' 't58' made a triangle then by 'r99' made a horizontal straight line and 't58' a triangle is drawn. As to make a head, made a square and then as a body be made a rectangle. Using circle wheels are drawn. Then exit.

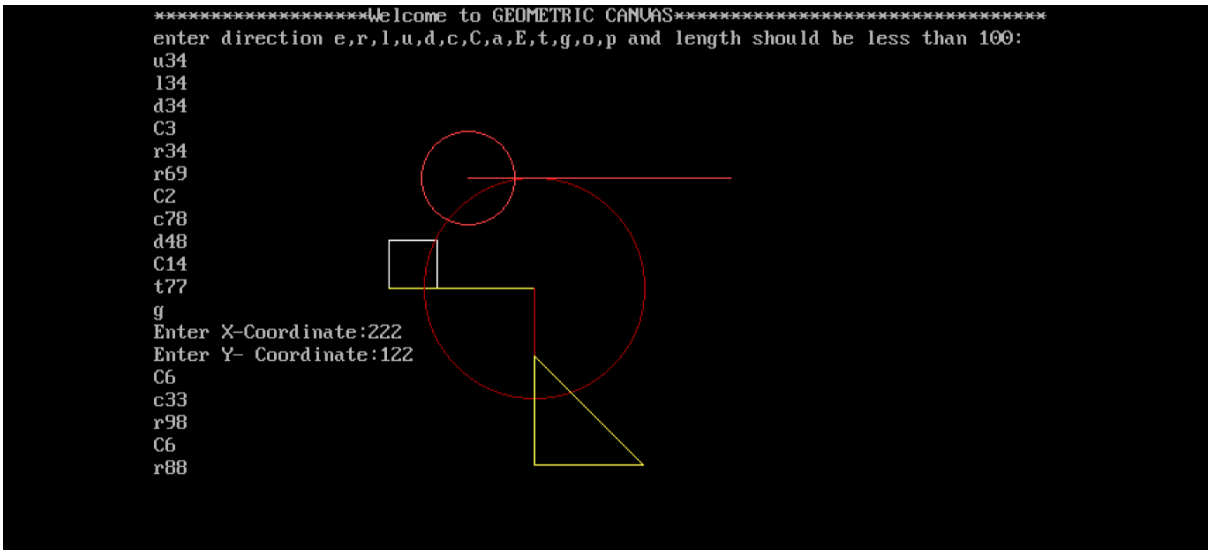

Fig 9: Sample output to represent 'g' function

In Figure 9 the pencil starts at coordinate(200,200)and moves 34 units upwards,34 units towards left,34 units downwards [WHITE] Color is changed to [yellow] and 34 units right move then 69 units right move is made. Next to that color is changed[red] and circle is drawn,then downward move is made from the centre of the circle and triangle[yellow] of measure (t77) is drawn. Input of coordinate is accepted and accordingly circle and line are drawn[red].

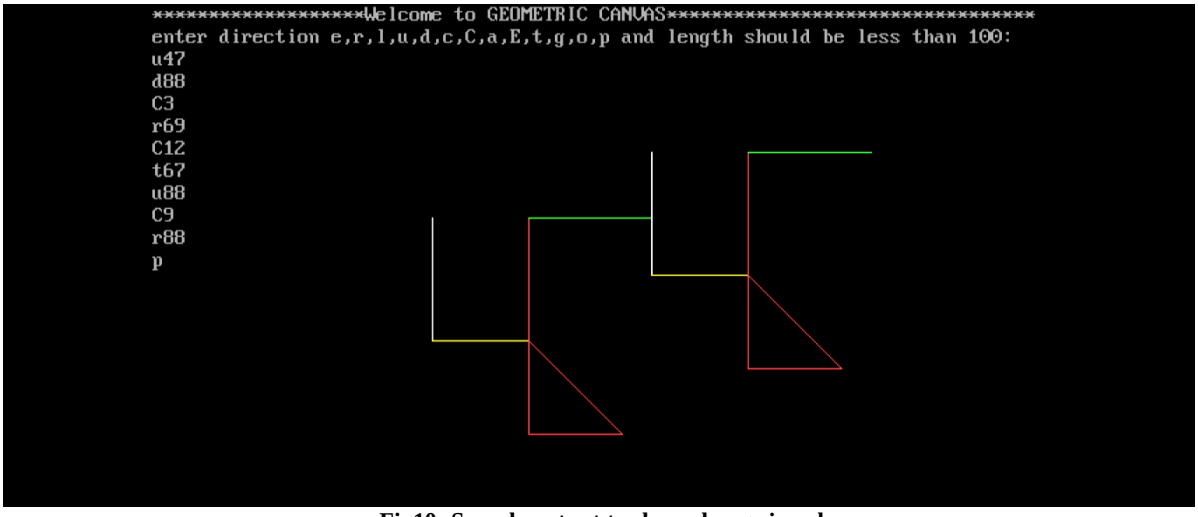

Fig10: Sample output to show change in colors

In Figure 10as by default the color of the pencil is white, upward and downward lines are drawn from the user defined coordinates. Then color is changed[yellow]and pencil is made move towards right then with changed color triangle is drawn[red]and upward move is made and finally right move is made [green].With the 'p' command all the moves previously performed are repeated and shown in the final step.

4. TESTING

To determine whether the final product behaves in accordance to the requirements, the testing phase is used. It involves using input data sets, also known as 'test cases', and evaluating the output for the corresponding input case.

The input set usually covers boundary conditions and certain situation where the program is known to deliver inconsistent output. Validation means checking the quality of software in both simulated and live environments. System validation ensures that the user can in fact match his/her claims, especially system performance. After a particular period of time, all the errors and failures that are detected are documented finally and are debugged and the program is finally designed as per the requirements of the user before its release. Errors are prone to occur in any phase of software development. In spite of employing error detection and removal methods, there remains the probability that certain errors may go unnoticed/ undetected. Also errors generated in one phase of

development may transition to the next phase without being detected. Such errors when detected at the final stage lead to unnecessary backtracking. Hence it is important to test the phases using concrete test cases to cover all possible failure conditions.

Testing objectives are:

- To determine whether program can take the required input specified by us or not.
- Whether any invalid input lead to program crash or not.
- Reaction of the output in case of invalid

4.1 Unit Testing

Any Program when developed is divided into components known as modules. Each Module deals with a particular functionality that is to be achieved by the program. Modules are developed in isolation to each other. There is a possibility that an error/bug/fault may occur during the coding or development phase of the module. Therefore, it is essential that individual modules also be checked for errors. The process of testing the independent modules is known as Unit Testing. Unit Testing has been done in the program and successfully processed. In this study, number of test cases have been taken and got successful output accordingly.

5. LIMITATIONS

- This project does not include the mobile application feature.
- This project does not include the feature to get the names of the shape.
- This project limits web access. The software does not work on rotating lines by using angle.

6. FUTURE SCOPE

- Description of shapes will be provided to learn shapes and their use in real life.
- The project can be implemented for web & mobile applications.
- More commands will be added according to need of kids.
- This project will have functionality to go to the coordinates using click.

7. CONCLUSION

This study has been able to complete project successfully to full satisfaction. As proposed, this study has created a basic platform for designing geometric and able to generate a code for the system. In the course of completion obtained a sound knowledge over general programming logic and C programming environment. The final phase which is the generation of the code is successfully completed and tested. Though this is an academic project but future enhancements can be made to this project by developing a GUI to help user to create the different geometric of choice.

8. REFERENCES

[1] https://en.wikipedia.org/wiki/Computational_geomety
[2] http://dictionary.sensagent.com
[3] Daniel Watt, Learning With Logo, McGraw Hill, ISBN 0-07-068570-3.
[4] http://people.csail.mit.edu/fredo/Depiction/1_Introduction/reviewGraphics.pdf
[5] ISRD Group, 2006. Computer Graphics, ISBN 0070593760.