**1. Introduction**

**1.1 Motivation**

Mathematics is a common subject which we keep learning right from our kindergarten till higher studies which include many equations which can represented graphically. The graphs of the equations are used to analyze and solve the equations. The main idea behind the application had taken birth when we ourselves had faced the problem to draw the graphs of any given mathematical equations to solve them. So we have designed an application which takes the mathematical equation as input and displays the graph in Cartesian plane .

**1.2 Objective of the Project**

The main Objective while implementing this application was to simplify drawing graphs of any mathematical equations and there by the analysis of those mathematical equations is made easy by judging their respective graphs.

**1.3 Tools**

This application has been totally developed on C++ platform by using 'graphics.h' exclusively for the implementation of Graphical User Interface. The IDE we used was Dev C++. The application can be run on any computer with basic specifications.

**2. Software and Tools**

**2.1.1 C++**

C++ is a statically-typed, free-form, compiled, multi-paradigm, intermediate-level general-purpose middle-level programming language.” In simple terms, C++ is a sophisticated, efficient and a general-purpose programming language based on C. It was developed by [Bjarne Stroustrup](http://www.stroustrup.com/" \o "Bjarne Stroustrup) in 1979.

Many of today’s operating systems, system drivers, browsers and games use C++ as their core language. This makes C++ one of the most popular languages today. Since it is an enhanced/extended version of [C programming](https://www.programiz.com/c-programming) language, C and C++ are often denoted together as C/C++.

**2.1.2 Features of C++**

* **C++ is an open ISO-standardized language.**   
  For a time, C++ had no official standard and was maintained by a de-facto standard, however since 1998, C++ is standardized by a committee of the ISO. Their page may be accessed [here](http://www.open-std.org/jtc1/sc22/wg21/).
* **C++ is a compiled language.**C++ compiles directly to a machine's native code, allowing it to be one of the fastest languages in the world, if optimized.
* **C++ is a strongly-typed unsafe language.**C++ is a language that expects the programmer to know what he or she is doing, but allows for incredible amounts of control as a result.
* **C++ supports both manifest and inferred typing.**   
  As of the latest C++ standard, C++ supports both manifest and inferred typing, allowing flexibility and a means of avoiding verbosity where desired.
* **C++ supports both static and dynamic type checking.**   
  C++ allows type conversions to be checked either at compile-time or at run-time, again offering another degree of flexibility. Most C++ type checking is, however, static.
* **C++ offers many paradigm choices.**C++ offers remarkable support for procedural, generic, and object-oriented programming paradigms, with many other paradigms being possible as well.
* **C++ is *portable.***As one of the most frequently used languages in the world and as an open language, C++ has a wide range of [compilers](http://www2.research.att.com/~bs/compilers.html) that run on many different platforms that support it. Code that exclusively uses C++'s standard library will run on many platforms with few to no changes.
* **C++ is upwards compatible with C**   
  C++, being a language that directly builds off C, is compatible with almost all C code. C++ can use C libraries with few to no modifications of the libraries' code.
* **C++ has incredible library support.**A search for "library" on the popular project-management website [SourceForge](http://www.sourceforge.net/) will yield over 3000 results for C++ libraries. A link to the results of the search may be found [here](http://sourceforge.net/directory/language:C%2B%2B/?q=library).

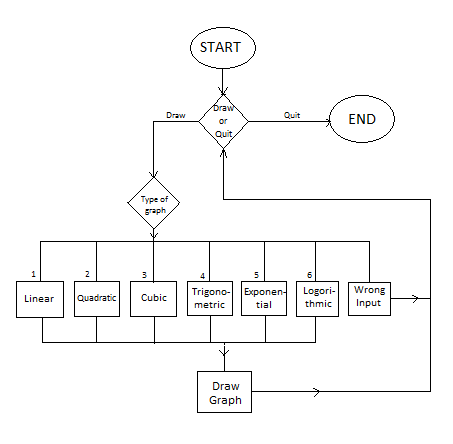
**2.2 <graphics.h> in C++**

C++ graphics using <graphics.h> functions can be used to draw different shapes, display text in different fonts, change [colors](https://www.programmingsimplified.com/c/graphics.h/colors) and many more. Using functions of <graphics.h> in Dev C++ you can make graphics programs, animations, projects, and games. You can draw circles, lines, rectangles, bars and many other geometrical figures. You can change their colors using the available functions and fill them. Following is a list of functions of graphics.h header file.

* void [arc](https://www.cs.colorado.edu/~main/bgi/doc/arc.html) (int x, int y, int stangle, int endangle, int radius);
* void [bar](https://www.cs.colorado.edu/~main/bgi/doc/bar.html) (int left, int top, int right, int bottom);
* void [bar3d](https://www.cs.colorado.edu/~main/bgi/doc/bar3d.html) (int left, int top, int right, int bottom, int depth, int topflag);
* ostringstream [bgiout](https://www.cs.colorado.edu/~main/bgi/doc/bgiout.html);
* void [circle](https://www.cs.colorado.edu/~main/bgi/doc/circle.html) (int x, int y, int radius);
* void [cleardevice](https://www.cs.colorado.edu/~main/bgi/doc/cleardevice.html) (void);
* void [clearmouseclick](https://www.cs.colorado.edu/~main/bgi/doc/clearmouseclick.html)(int kind);
* void [clearviewport](https://www.cs.colorado.edu/~main/bgi/doc/clearviewport.html) (void);
* void [closegraph](https://www.cs.colorado.edu/~main/bgi/doc/closegraph.html) (int window=ALL\_WINDOWS);
* int [converttorgb](https://www.cs.colorado.edu/~main/bgi/doc/wincolor.html) (int color);
* void [delay](https://www.cs.colorado.edu/~main/bgi/doc/delay.html) (int millisec);
* void [detectgraph](https://www.cs.colorado.edu/~main/bgi/doc/detectgraph.html) (int \*graphdriver, int \*graphmode);
* void [drawpoly](https://www.cs.colorado.edu/~main/bgi/doc/drawpoly.html) (int numpoints, int \*polypoints);
* void [ellipse](https://www.cs.colorado.edu/~main/bgi/doc/ellipse.html) (int x, int y, int stangle, int endangle, int xradius, int yradius);
* void [fillellipse](https://www.cs.colorado.edu/~main/bgi/doc/fillellipse.html) (int x, int y, int xradius, int yradius);
* void [fillpoly](https://www.cs.colorado.edu/~main/bgi/doc/fillpoly.html) (int numpoints, int \*polypoints);
* void [floodfill](https://www.cs.colorado.edu/~main/bgi/doc/floodfill.html) (int x, int y, int border);
* int [getactivepage](https://www.cs.colorado.edu/~main/bgi/doc/getactivepage.html) (void);
* void [getarccoords](https://www.cs.colorado.edu/~main/bgi/doc/getarccoords.html) (struct arccoordstype \*arccoords);
* void [getaspectratio](https://www.cs.colorado.edu/~main/bgi/doc/getaspectratio.html) (int \*xasp, int \*yasp);
* int [getbkcolor](https://www.cs.colorado.edu/~main/bgi/doc/getbkcolor.html) (void);
* int [getch](https://www.cs.colorado.edu/~main/bgi/doc/getch.html) (void);
* int [getcolor](https://www.cs.colorado.edu/~main/bgi/doc/getcolor.html) (void);
* int [getcurrentwindow](https://www.cs.colorado.edu/~main/bgi/doc/getcurrentwindow.html) (void);
* struct palettetype\* [getdefaultpalette](https://www.cs.colorado.edu/~main/bgi/doc/getdefaultpalette.html) (void);
* int [getdisplaycolor](https://www.cs.colorado.edu/~main/bgi/doc/getdisplaycolor.html) (int color);
* char\* [getdrivername](https://www.cs.colorado.edu/~main/bgi/doc/getdrivername.html) (void);
* void [getfillpattern](https://www.cs.colorado.edu/~main/bgi/doc/getfillpattern.html) (char \*pattern);
* void [getfillsettings](https://www.cs.colorado.edu/~main/bgi/doc/getfillsettings.html) (struct fillsettingstype \*fillinfo);
* int [getgraphmode](https://www.cs.colorado.edu/~main/bgi/doc/getgraphmode.html) (void);
* void [getimage](https://www.cs.colorado.edu/~main/bgi/doc/getimage.html) (int left, int top, int right, int bottom, void \*bitmap);
* void [getlinesettings](https://www.cs.colorado.edu/~main/bgi/doc/getlinesettings.html) (struct linesettingstype \*lineinfo);
* int [getmaxcolor](https://www.cs.colorado.edu/~main/bgi/doc/getmaxcolor.html) (void);
* int [getmaxmode](https://www.cs.colorado.edu/~main/bgi/doc/getmaxmode.html) (void);
* int [getmaxheight](https://www.cs.colorado.edu/~main/bgi/doc/getmaxheight.html) (void);
* int [getmaxwidth](https://www.cs.colorado.edu/~main/bgi/doc/getmaxwidth.html) (void);
* int [getmaxx](https://www.cs.colorado.edu/~main/bgi/doc/getmaxx.html) (void);
* int [getmaxy](https://www.cs.colorado.edu/~main/bgi/doc/getmaxy.html) (void);
* char\* [getmodename](https://www.cs.colorado.edu/~main/bgi/doc/getmodename.html) (int mode\_number);
* void [getmoderange](https://www.cs.colorado.edu/~main/bgi/doc/getmoderange.html) (int graphdriver, int \*lomode, int \*himode);
* void [getmouseclick](https://www.cs.colorado.edu/~main/bgi/doc/getmouseclick.html)(int kind, int& x, int& y);
* void [getpalette](https://www.cs.colorado.edu/~main/bgi/doc/getpalette.html) (struct palettetype \*palette);
* int [getpalettesize](https://www.cs.colorado.edu/~main/bgi/doc/getpalettesize.html) (void);
* int [getpixel](https://www.cs.colorado.edu/~main/bgi/doc/getpixel.html) (int x, int y);
* void [gettextsettings](https://www.cs.colorado.edu/~main/bgi/doc/gettextsettings.html) (struct textsettingstype \*texttypeinfo);
* void [getviewsettings](https://www.cs.colorado.edu/~main/bgi/doc/getviewsettings.html) (struct viewporttype \*viewport);
* int [getvisualpage](https://www.cs.colorado.edu/~main/bgi/doc/getvisualpage.html) (void);
* int [getwindowheight](https://www.cs.colorado.edu/~main/bgi/doc/getwindowheight.html) (void);
* int [getwindowwidth](https://www.cs.colorado.edu/~main/bgi/doc/getwindowwidth.html) (void);
* int [getx](https://www.cs.colorado.edu/~main/bgi/doc/getx.html) (void);
* int [gety](https://www.cs.colorado.edu/~main/bgi/doc/gety.html) (void);
* void [graphdefaults](https://www.cs.colorado.edu/~main/bgi/doc/graphdefaults.html) (void);
* char\* [grapherrormsg](https://www.cs.colorado.edu/~main/bgi/doc/grapherrormsg.html) (int errorcode);
* int [graphresult](https://www.cs.colorado.edu/~main/bgi/doc/graphresult.html)(void);
* unsigned [imagesize](https://www.cs.colorado.edu/~main/bgi/doc/imagesize.html) (int left, int top, int right, int bottom);
* void [initgraph](https://www.cs.colorado.edu/~main/bgi/doc/initgraph.html) (int \*graphdriver, int \*graphmode, char \*pathtodriver);
* int [initwindow](https://www.cs.colorado.edu/~main/bgi/doc/initwindow.html) (int width, int height, const char\* title="Windows BGI", int left=0, int top=0, bool dbflag=false, bool closeflag=true);
* int [installuserdriver](https://www.cs.colorado.edu/~main/bgi/doc/installuserdriver.html) (char \*name, int huge (\*detect)(void));
* int [installuserfont](https://www.cs.colorado.edu/~main/bgi/doc/installuserfont.html) (char \*name);
* bool [ismouseclick](https://www.cs.colorado.edu/~main/bgi/doc/ismouseclick.html)(int kind);
* int [kbhit](https://www.cs.colorado.edu/~main/bgi/doc/kbhit.html) (void);
* void [line](https://www.cs.colorado.edu/~main/bgi/doc/line.html) (int x1, int y1, int x2, int y2);
* void [linerel](https://www.cs.colorado.edu/~main/bgi/doc/linerel.html) (int dx, int dy);
* void [lineto](https://www.cs.colorado.edu/~main/bgi/doc/lineto.html) (int x, int y);
* int [mousex](https://www.cs.colorado.edu/~main/bgi/doc/mousex.html) (void);
* int [mousey](https://www.cs.colorado.edu/~main/bgi/doc/mousey.html) (void);
* void [moverel](https://www.cs.colorado.edu/~main/bgi/doc/moverel.html) (int dx, int dy);
* void [moveto](https://www.cs.colorado.edu/~main/bgi/doc/moveto.html) (int x, int y);
* void [outtext](https://www.cs.colorado.edu/~main/bgi/doc/outtext.html) (char \*textstring);
* void [outtextxy](https://www.cs.colorado.edu/~main/bgi/doc/outtextxy.html) (int x, int y, char \*textstring);
* void [pieslice](https://www.cs.colorado.edu/~main/bgi/doc/pieslice.html) (int x, int y, int stangle, int endangle, int radius);
* void [printimage](https://www.cs.colorado.edu/~main/bgi/doc/printimage.html) (const char\* title=NULL, double width\_inches=7,double order\_left\_inches=0.75, double border\_top\_inches=0.75, int left=0, int right=0, int right=INT\_MAX, int bottom=INT\_MAX);
* void [putimage](https://www.cs.colorado.edu/~main/bgi/doc/putimage.html) (int left, int top, void \*bitmap, int op);
* void [putpixel](https://www.cs.colorado.edu/~main/bgi/doc/putpixel.html) (int x, int y, int color);
* void [readimagefile](https://www.cs.colorado.edu/~main/bgi/doc/readimagefile.html) (const char\* filename=NULL,int left=0, int top=0, int right=INT\_MAX, int bottom=INT\_MAX);
* void [rectangle](https://www.cs.colorado.edu/~main/bgi/doc/rectangle.html) (int left, int top, int right, int bottom);
* int [registerbgidriver](https://www.cs.colorado.edu/~main/bgi/doc/registerbgidriver.html) (void (\*driver)(void));
* int [registerbgifont](https://www.cs.colorado.edu/~main/bgi/doc/registerbgifont.html) (void (\*font)(void));
* void [registermousehandler](https://www.cs.colorado.edu/~main/bgi/doc/registermousehandler.html) (int kind, void h(int, int));
* void [restorecrtmode](https://www.cs.colorado.edu/~main/bgi/doc/restorecrtmode.html) (void);
* [RGB functions:](https://www.cs.colorado.edu/~main/bgi/doc/wincolor.html)
* COLOR(r,g,b),
* RED\_VALUE(v), GREEN\_VALUE(v), BLUE\_VALUE(v),
* IS\_BGI\_COLOR(v), IS\_RGB\_COLOR(v)
* void [sector](https://www.cs.colorado.edu/~main/bgi/doc/sector.html) (int x, int y, int stangle, int endangle, int xradius, int yradius);
* void [setactivepage](https://www.cs.colorado.edu/~main/bgi/doc/setactivepage.html) (int page);
* void [setallpalette](https://www.cs.colorado.edu/~main/bgi/doc/setallpalette.html) (struct palettetype \*palette);
* void [setaspectratio](https://www.cs.colorado.edu/~main/bgi/doc/setaspectratio.html) (int xasp, int yasp);
* void [setbkcolor](https://www.cs.colorado.edu/~main/bgi/doc/setbkcolor.html) (int color);
* void [setcolor](https://www.cs.colorado.edu/~main/bgi/doc/setcolor.html) (int color);
* void [setcurrentwindow](https://www.cs.colorado.edu/~main/bgi/doc/setcurrentwindow.html) (int window);
* void [setmousequeuestatus](https://www.cs.colorado.edu/~main/bgi/doc/setmousequeuestatus.html)(int kind, bool status=true);
* void [setfillpattern](https://www.cs.colorado.edu/~main/bgi/doc/setfillpattern.html) (char \*upattern, int color);
* void [setfillstyle](https://www.cs.colorado.edu/~main/bgi/doc/setfillstyle.html) (int pattern, int color);
* unsigned [setgraphbufsize](https://www.cs.colorado.edu/~main/bgi/doc/setgraphbufsize.html) (unsigned bufsize);
* void [setgraphmode](https://www.cs.colorado.edu/~main/bgi/doc/setgraphmode.html) (int mode);
* void [setlinestyle](https://www.cs.colorado.edu/~main/bgi/doc/setlinestyle.html) (int linestyle, unsigned upattern, int thickness);
* void [setpalette](https://www.cs.colorado.edu/~main/bgi/doc/setpalette.html) (int colornum, int color);
* void [setrgbpalette](https://www.cs.colorado.edu/~main/bgi/doc/setrgbpalette.html) (int colornum, int red, int green, int blue);
* void [settextjustify](https://www.cs.colorado.edu/~main/bgi/doc/settextjustify.html) (int horiz, int vert);
* void [settextstyle](https://www.cs.colorado.edu/~main/bgi/doc/settextstyle.html) (int font, int direction, int charsize);
* void [setusercharsize](https://www.cs.colorado.edu/~main/bgi/doc/setusercharsize.html) (int multx, int divx, int multy, int divy);
* void [setviewport](https://www.cs.colorado.edu/~main/bgi/doc/setviewport.html) (int left, int top, int right, int bottom, int clip);
* void [setvisualpage](https://www.cs.colorado.edu/~main/bgi/doc/setvisualpage.html) (int page);
* void [setwritemode](https://www.cs.colorado.edu/~main/bgi/doc/setwritemode.html) (int mode);
* int [showerrorbox](https://www.cs.colorado.edu/~main/bgi/doc/showerrorbox.html) (const char \*message);
* int [swapbuffers](https://www.cs.colorado.edu/~main/bgi/doc/swapbuffers.html) (void);
* int [textheight](https://www.cs.colorado.edu/~main/bgi/doc/textheight.html) (char \*textstring);
* int [textwidth](https://www.cs.colorado.edu/~main/bgi/doc/textwidth.html) (char \*textstring);
* void [writeimagefile](https://www.cs.colorado.edu/~main/bgi/doc/writeimagefile.html) (const char\* filename=NULL,double width\_inches=7, double border\_left\_inches=0.75, double border\_top\_inches=0.75,int left=0, int top=0, int right=INT\_MAX, int bottom=INT\_MAX);

**3. System Design**

**3.1 Flow Chart**

****

**Fig 3.1 . Flowchart representing the operations carried out**

**4. Implementation**

**4.1 Code Structure**

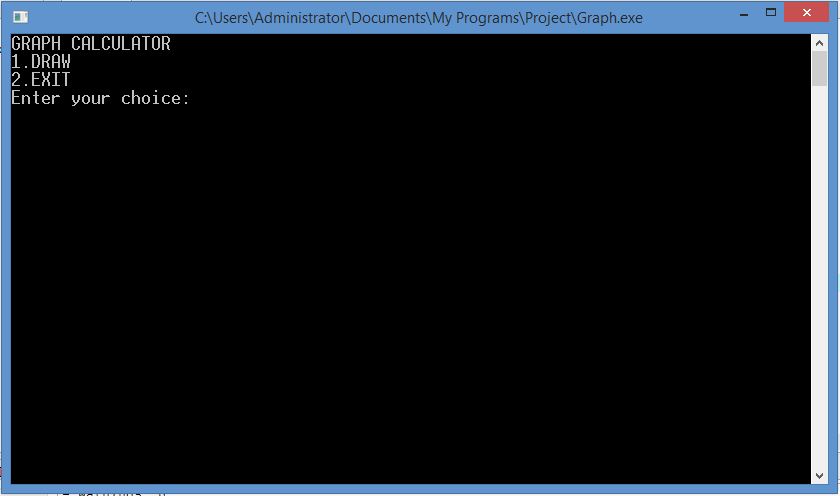
The first thing required was the <graphics.h>. So we have downloaded <graphics.h> header file from the internet and included in the libraries of our IDE i.e., Dev C++. Then this header file was also included into our program with the statement "#include<graphics.h>" in the code. To obtain Graphical User Interface our Console Application's linker options were changed as per the requirements. Then we have written code to develop different types of graphs using different functions for each and every type of graph. The list of functions and their description is given below.

1. void axes() : Draws the axes.
2. void point(int xd,int yd) : Displays the coordinates of the graph using pointer device.
3. int choosecolor() : Displays menu to choose graph color and sets the color as per the input given
4. void sine(int color) : Draws Trigonometric sine graph.
5. void asine(int color) : Draws Trigonometric sine inverse graph.
6. void sineh(int color) : Draws Trigonometric sine hyperbolic graph.
7. void asineh(int color) : Draws Trigonometric sine inverse hyperbolic graph.
8. void cosine(int color) : Draws Trigonometric cosine graph.
9. void acosine(int color) : Draws Trigonometric cosine inverse graph.
10. void cosineh(int color) : Draws Trigonometric cosine hyperbolic graph.
11. void acosineh(int color) : Draws Trigonometric cosine inverse hyperbolic graph.
12. void tangent(int color) : Draws Trigonometric tangent graph.
13. void atangent(int color) : Draws Trigonometric tangent inverse graph.
14. void tangenth(int color) : Draws Trigonometric tangent hyperbolic graph.
15. void atangenth(int color) :Draws Trigonometric tangent inverse hyperbolic graph.
16. void Sqrt(int color) : Draws square root graph.
17. void Cbrt(int color) : Draws cube root graph.
18. void Log(int color) : Draws Logarithm base 10 graph.
19. void ln(int color) : Draws Logarithm base e graph.
20. void Exp(int color) : Draws e^x graph.
21. void Pow(int k,int color) : Draws x^k graph.
22. void lin(int color) : Draws Linear equation graph.
23. void quad(int color) : Draws Quadratic equation graph.
24. void cub(int color) : Draws Cubic equation graph.

The main() function contains the menu format in a hierarchical switch case statements with above defined function calling statements.

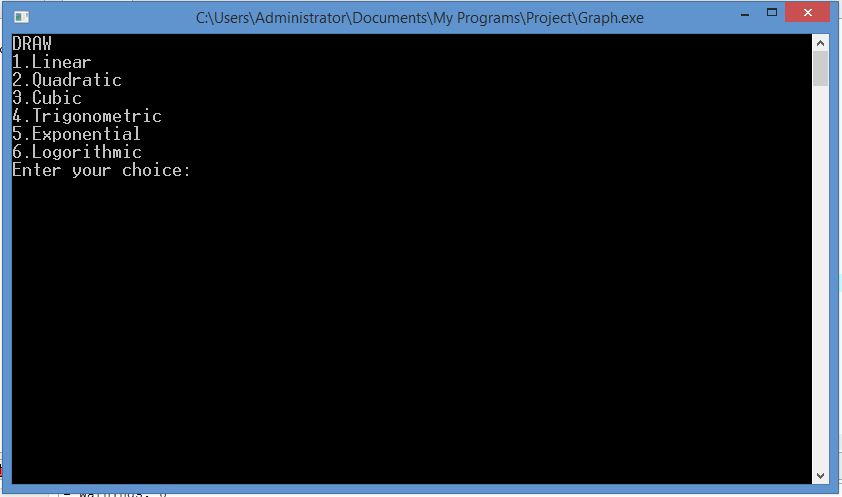
**4.2 Execution**

1. The execution of the program starts by asking the user either to Draw or Exit. Choosing Exit terminates the program.



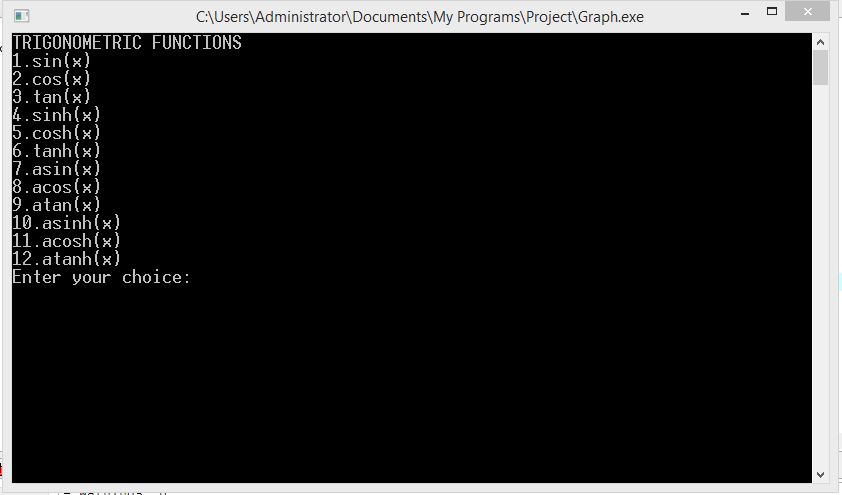
**Fig 4.1. Main Menu**

1. After choosing Draw option the user is asked to choose the type of the graph he wants.

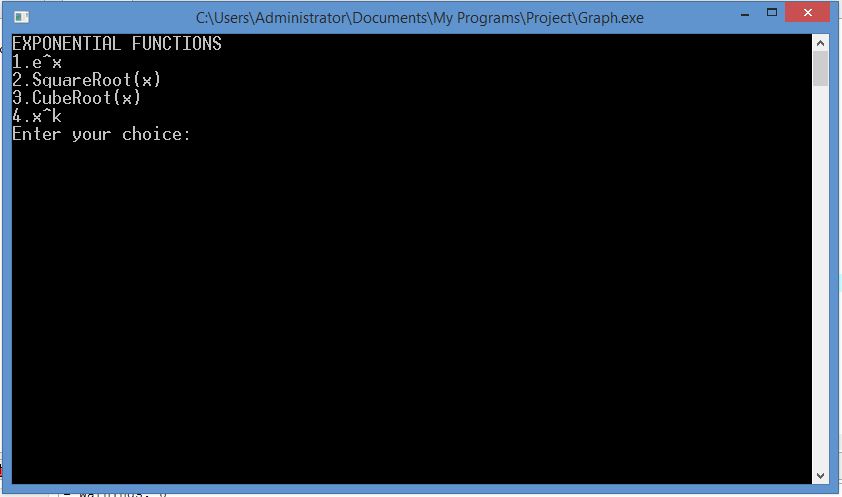
****

**Fig 4.2. Menu of different types of graphs**

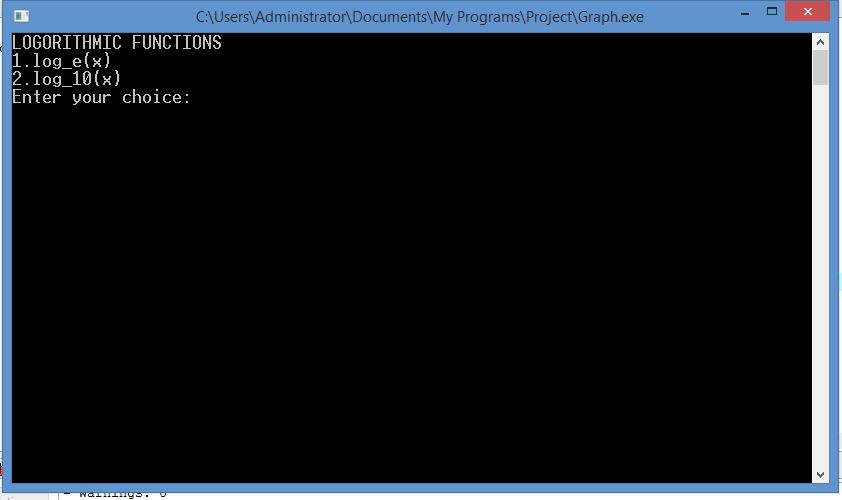
1. There are sub menus in Trigonometric, Exponential, Logarithmic types of graphs.



**Fig 4.3. Trigonometric Functions Menu**

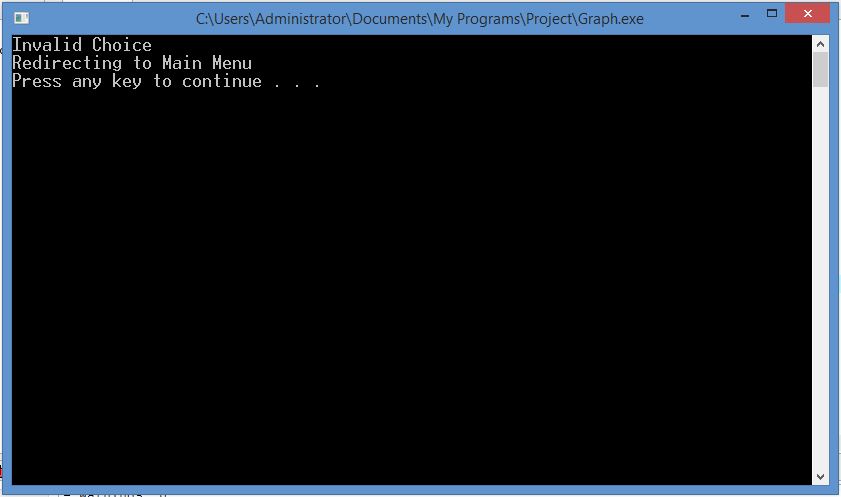


**Fig 4.4. Exponential Functions Menu**



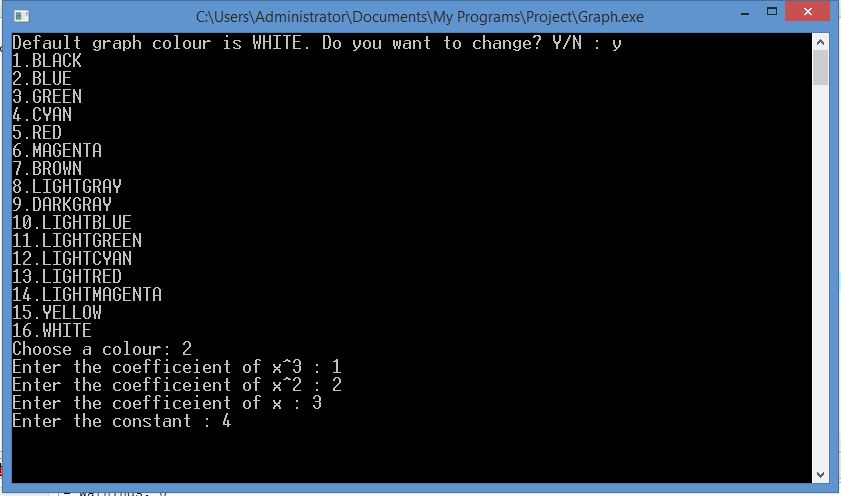
**Fig 4.5. Logarithmic Functions Menu**

1. If wrong option is entered in any menu, program is redirected to main menu.



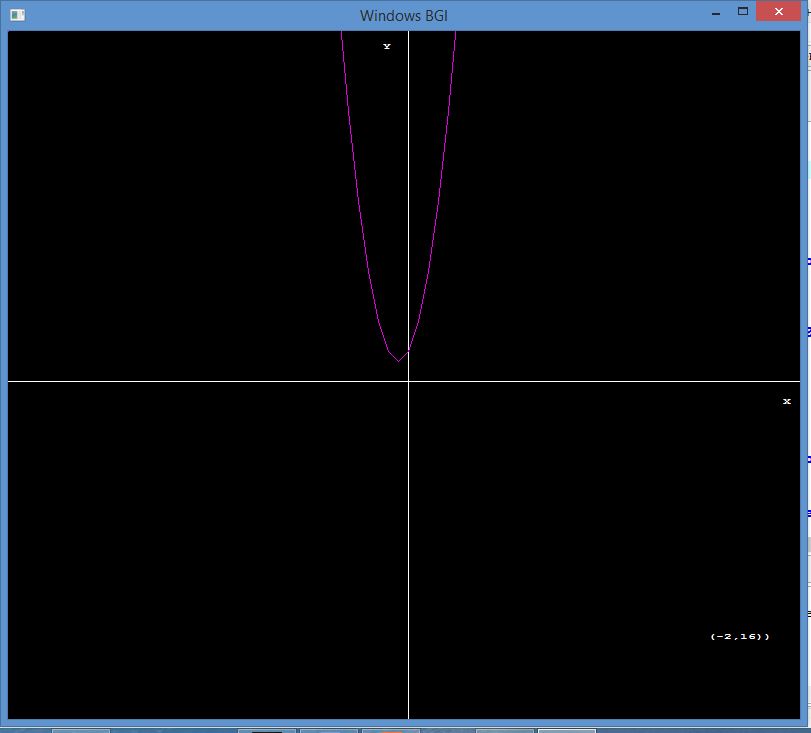
**Fig 4.6. Entry of invalid choice**

1. After selecting which graph to be drawn, it takes all the required information to draw the graph from the user.

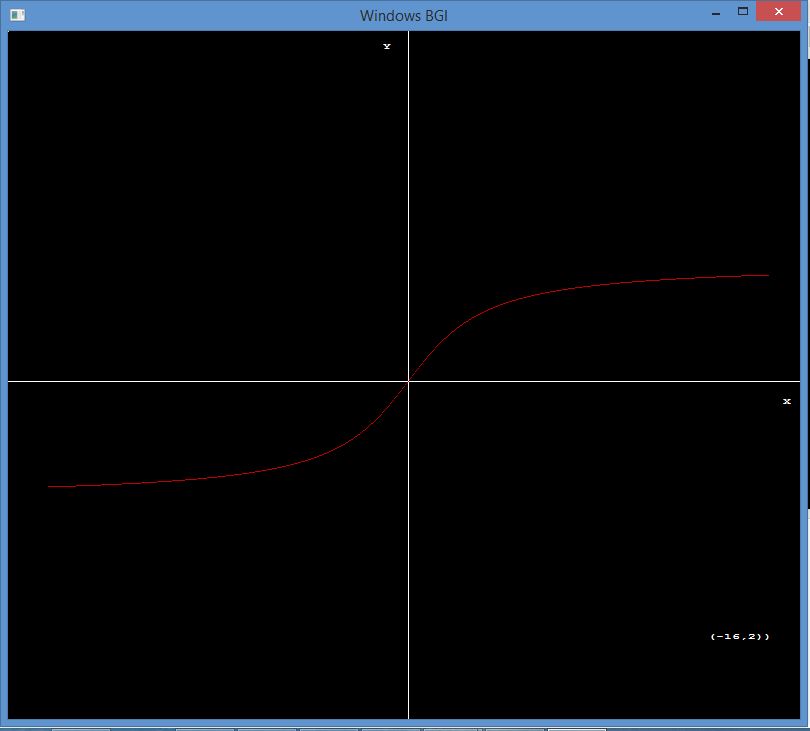


**Fig 4.7. Input of required information**

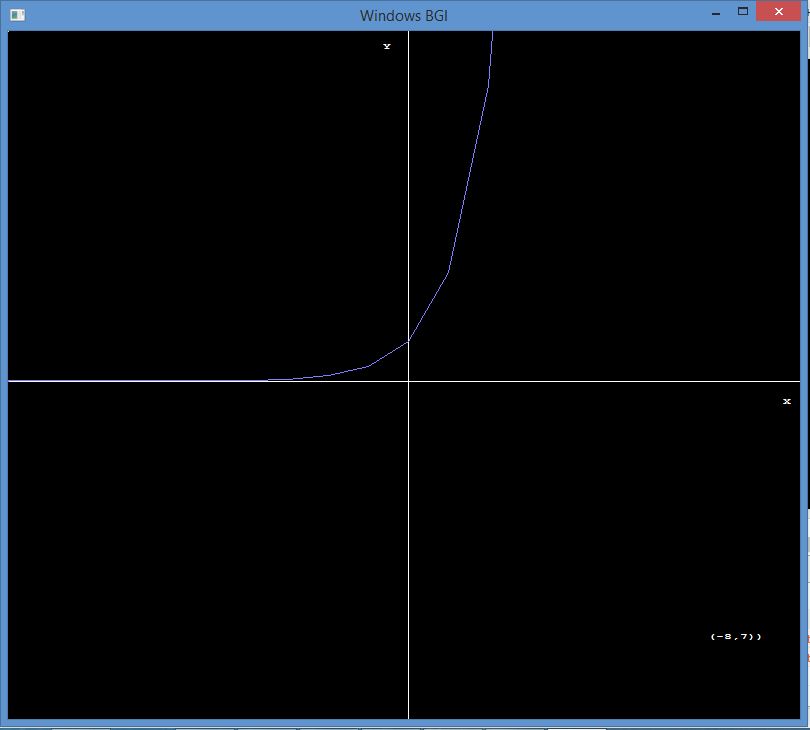
1. The Graph is Drawn. Some sample Graphs are shown below.



**Fig 4.8. Graph of a Quadratic Equation.**

****

**Fig 4.9. Graph of Inverse Tangent Function**

****

**Fig 4.10 Graph of e^x**

**5. Conclusion**

Using this application will make life very simple for students who study Mathematics mainly for higher studies of Mathematics, Physics and Chemistry. Various concepts and phenomena in Physics and Chemistry are explained mathematically and represented graphically. In each and every such concept or phenomenon this application is useful which displays the graph by taking the mathematical equation as input.

The main Objective while implementing this application was to simplify drawing graphs of any mathematical equations and there by the analysis of those mathematical equations is made easy by judging their respective graphs. We designed the program in such a way that it perfectly matched our objectives.

**6. Future Scope**

The next version of this application will have many more features like :

* Two different graphs can be drawn and compared at a time.
* The front end of the application can be designed in a web page.
* The application can be made usable online.
* The application can be released in Android version.

**7. Bibliography**

1. The C++ Programming Language by Bjarne Stroustrup
2. www.koolplot.com
3. en.wikipedia.org
4. www.cplusplus.com
5. www.geeksforgeeks.com
6. www.quora.com