**Practical No 1**

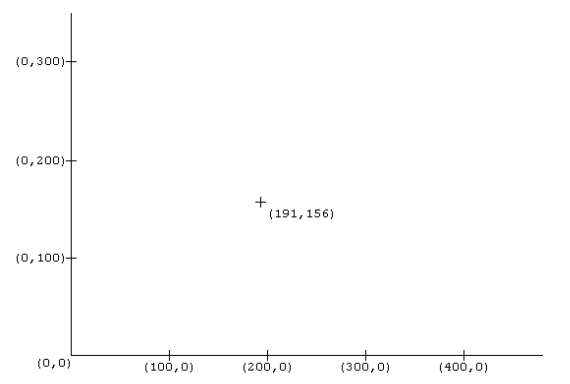
**A) Prepare well formatted document covering graphics coordinates system and selective functions from Graphics.h**

**Aim: To study about the graphics coordinates system and selective functions from Graphics.h**

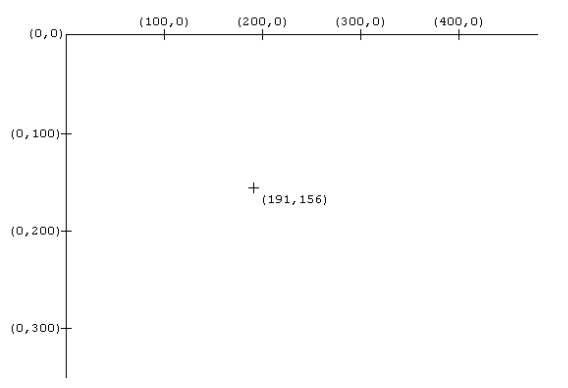
**Theory:**

**I] Co-ordinate System:**

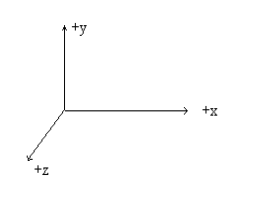
In Co-ordinate system there are 2 dimensional cartesian which consist of an origin and two axes, which is at the right angle to each other’s and an intersection at a point which is called as the origin. Each point on the plane containing the coordinates which is uniquely identified by two numbers. First number represent the signed horizontal distance from the vertical axis and second number represent the signed vertical distance from the horizontal axis. Horizontal dimension increases to the right from the origin whereas the vertical dimension upward from the origin.



Computer screens typically use a coordinate system which also consists of an origin and two axes, at right angles to each other, intersecting at a point called the origin which is located at top left corner of the screen. Each point on the plane containing the coordinates is uniquely identified by two numbers. The first represents the signed horizontal distance from the vertical axis, and the second represents the signed vertical distance from the horizontal axis. The horizontal dimension increases to the right from the origin, and the vertical dimension downward from the origin.



The 3-dimensional Cartesian coordinate system consists of an origin and three axes, at right angles to each other, intersecting at a point called the origin. Each point on the space containing the coordinates is uniquely identified by three numbers. The first represents the signed horizontal distance from the vertical axis, measured parallel to the X axis, the second represents the signed vertical distance from the horizontal axis, measured parallel to the Y axis, and the third represents the signed horizontal distance from the vertical axis, measured parallel to the Z axis. The X dimension increases to the right from the origin, and the Y dimension upward from the origin, and the Z dimension increases outward from the surface of the image as shown. This actually describes a right-handed coordinate system, in which a right-handed thread along the Z axis will travel toward increasing Z dimensions when rotated in a counter-clockwise rotation.



**II] Basic Library Functions in Graphics.h**

Functions in graphics.h can be used to draw different shapes, display text in different fonts, change colors and many more. Graphics programs, animations, projects and games can be implemented using inbuilt functions. We can draw circles, lines, rectangles, bars and many other geometrical figures. We can change their colors using the available functions and fill them.

**Following is a list of functions of graphics.h header file.**

**1) getpixel:** getpixel function returns the color of pixel present at location (x, y).

**Declaration: -** int getpixel (int x, int y);

**2) putpixel:** putpixel function plots a pixel at location (x, y) of specified color.

**Declaration: -** void putpixel (int x, int y, int color);

**3) line:** line function is used to draw a line from a point (x1, y1) to point (x2, y2) i.e. (x1, y1) and (x2, y2) are end points of the line.

**Declaration: -** void line (int x1, int y1, int x2, int y2);

**4) circle:** Circle function is used to draw a circle with centre (x, y) and third parameter specifies the radius of the circle.

**Declaration: -** void circle (int x, int y, int radius);

**5) closegraph:** closegraph function closes the graphics mode, deallocates all memory allocated by graphics system and restores the screen to the mode it was in before you called initgraph.

**Declaration: -** void closegraph ();

**6) drawpoly:** This function is used to draw polygons i.e., triangle, rectangle, pentagon, hexagon etc.

**Declaration: -** void drawpoly (int num, int \*polypoints);

Where, num indicates (n+1) number of points where n is the number of vertices in a polygon, polypoints points to a sequence of (n\*2) integers. Each pair of integers gives x and y coordinates of a point on the polygon. We specify (n+1) points as first point coordinates should be equal to (n+1) th to draw a complete figure.

**eg:** int points [] = {320, 150, 420, 300, 250, 300, 320, 150};

The above example will draw a triangle where (320, 150), (420, 300) and (250, 300) are coordinates of triangle, last point (320, 150) in array is same as first.

**7) ellipse:** This function is used to draw an ellipse (x, y) are coordinates of centre of the ellipse, stangle is the starting angle, end angle is the ending angle, and fifth and sixth parameters specifies the X and Y radius of the ellipse. To draw a complete ellipse strangles and end angle should be 0 and 360 respectively.

**Declarations: -** void ellipse (int x, int y, int stangle, int endangle, int xradius, int yradius);

**8) getmaxx:** This function returns the maximum X coordinate for current graphics mode and driver.

**Declaration: -** int getmaxx ();

**9) getmaxy:** This function returns the maximum Y coordinate for current graphics mode and driver.

**Declaration: -** int getmaxy ();

**10) setcolor:** In Graphics each color is assigned a number. Total 16 colors are available. For

**Declaration: -** void setcolor (int color);

**Conclusion: We have studied the graphics coordinates system and selective functions from Graphics.h**