**Practical No 1**

**B) Creation of smiley like object using simple inbuilt graphic functions from graphics.h**

**Aim: To create a smiley like object using simple inbuilt graphic functions from graphics.h**

**Theory:**

* Graphics plays an important part in every application you develop.
* graphics.h library is frequently used to make graphics in C language.
* The graphics.h header file provides access to a simple graphics library that makes it possible to draw lines, rectangles, ovals, arcs, polygons, images, and strings on a graphical window.
* The Graphics Object takes much of the pain out of graphics drawing by abstracting away all the problems of dealing with different display devices and screens resolutions.
* Making a circle and an ellipse in cpp can be done easily.
* We first initialize a graph with two parameters and a path to the "BGI" folder in our system.
* After that, we will call the function called circle () with three numbers as the coordinates of the center and radius.
* So, this function will create a circle with a center with the given radius.
* To make an ellipse on the screen, all we need to do is call the ellipse () function with six numbers as the coordinates of the ellipse.
* These six co-ordinates decide the location of the ellipse, angles, and radius from X-axis and Y-axis.
* To use these functions in your program, we would need to include graphics.h file in your program.
* arc () is included in the header file called as graphics.h which draw the arc with center at (x, y) with given radius.
* You should also use getch () function to make the screen freeze.
* So, with the main function circle, ellipse and arc we draw the smiley like object.
* **Syntax:** #include<graphics.h>

**Conclusion: We created a smiley like object using simple inbuilt graphic functions from graphics.h**

**Code:**

#include <graphics.h>

#include <conio.h>

int main(void)

{

int gdriver = DETECT, gmode, errorcode;

int xmax, ymax;

initgraph(&gdriver, &gmode, "C:\\TC\\BGI");

xmax = getmaxx();

ymax = getmaxy();

circle(xmax/2,ymax/2,100);

ellipse(xmax/2, ymax/2, 220, 320, 50,70);

ellipse((xmax/2)-40,(ymax/2)-15,0,360,12,15);

arc((xmax/2)+40,(ymax/2)+10,40,130,20);

arc((xmax/2)+40,(ymax/2)+10,40,90,35);

arc((xmax/2)-40,(ymax/2)-15,90,150,30);

getch();

closegraph();

return 0;

}

**Output:**

