### Installation Notes:

1. Install Dev-C++. I installed from the [Version 4.9.9.2 Setup File](https://home.cs.colorado.edu/~main/bgi/dev-c++/devcpp-4.9.9.2_setup.exe).
2. Download [graphics.h](https://home.cs.colorado.edu/~main/bgi/dev-c++/graphics.h) to the include/ subdirectory of the Dev-C++ directories.
3. Download [libbgi.a](https://home.cs.colorado.edu/~main/bgi/dev-c++/libbgi.a) to the lib/ In order to use the WinBGIm subdirectory of the Dev-C++ directories.
4. Whenever you #include <graphics.h> in a program, you must instruct the linker to link in certain libraries. The command to do so from Dev-C++ is Alt-P. Choose the Parameters tab from the pop-up window and type the following into the Linker area:

-lbgi

-lgdi32

-lcomdlg32

-luuid

-loleaut32

-lole32

You can now compile and run programs that use the WinBGIm graphics library, such as this one that opens a small window, draws a circle and waits for the user to press a key:

#include <graphics.h>

int main( )

{

initwindow(400, 300, "First Sample");

circle(100, 50, 40);

while (!kbhit( ))

{

delay(200);

}

return 0;

}

void initgraph(int \*graphicsDriver, int \*graphicsMode, char \*driverDirectoryPath);

It initializes the graphics system by loading the passed graphics driver then changing the system into graphics mode. It also resets or initializes all graphics settings like color, palette, current position etc, to their default values. Below is the description of input parameters of initgraph function.

graphicsDriver : It is a pointer to an integer specifying the graphics driver to be used. It tells the compiler that what graphics driver to use or to automatically detect the drive. In all our programs we will use DETECT macro of graphics.h library that instruct compiler for auto detection of graphics driver.

graphicsMode : It is a pointer to an integer that specifies the graphics mode to be used. If \*gdriver is set to DETECT, then initgraph sets \*gmode to the highest resolution available for the detected driver.

driverDirectoryPath : It specifies the directory path where graphics driver files (BGI files) are located. If directory path is not provided, then it will search for driver files in current working directory directory. In all our sample graphics programs, you have to change path of BGI directory accordingly where you Turbo C++ compiler is installed.

#include<graphics.h>

#include<stdio.h>

void main(void) {

int gdriver = DETECT, gmode;

int x1 = 200, y1 = 200;

int x2 = 300, y2 = 300;

initgraph(&gdriver, &gmode, "c:\\turboc3\\bgi");

line(x1, y1, x2, y2);

closegraph();

}

#include<graphics.h>

#include<conio.h>

**void** main()

{

    Int gd=DETECT,gm;

    initgraph (&gd,&gm,"c:\\tc\\bgi");

    setbkcolor(GREEN);

    printf("\t\t\t\n\nLINE");

    line(50,40,190,40);

    printf("\t\t\n\n\n\nRECTANGLE");

    rectangle(125,115,215,165);

    printf("\t\t\t\n\n\n\n\n\n\nARC");

    arc(120,200,180,0,30);

    printf("\t\n\n\n\nCIRCLE");

    circle(120,270,30);

    printf("\t\n\n\n\nECLIPSE");

    ellipse(120,350,0,360,30,20);

    getch();

}

#include<graphics.h>

#include<conio.h>

void main()

{

    intgd=DETECT, gm, i, x, y;

    initgraph(&gd, &gm, "C:\\TC\\BGI");

    x=getmaxx()/3;

    y=getmaxx()/3;

    setbkcolor(WHITE);

    setcolor(BLUE);

    for(i=1;i<=8;i++)

          {

        setfillstyle(i,i);

        delay(20);

        circle(x, y, i\*20);

        floodfill(x-2+i\*20,y,BLUE);

    }

    getch();

    closegraph();

}

Color Description in C

setbkcolor sets the background to the color specified by the color or the number. The argument color may be a name or a number as given in the table below. (These symbolic names are defined in graphics.h). These colors can also be used to set textcolor (color of the text) or filling inside various shapes that you make in your program. We shall first learn about the color and their values and then we will learn it via the programs.

Color Numeric Value

BLACK 0

BLUE 1

GREEN 2

CYAN 3

RED 4

MAGENTA 5

BROWN 6

LIGHTGRAY 7

DARKGRAY 8

LIGHTBLUE 9

LIGHTGREEN 10

LIGHTCYAN 11

LIGHTRED 12

LIGHTMAGENTA 13

YELLOW 14

WHITE 15

## C graphics functions

* [arc](https://www.programmingsimplified.com/c/graphics.h/arc)
* [bar](https://www.programmingsimplified.com/c/graphics.h/bar)
* [bar3d](https://www.programmingsimplified.com/c/graphics.h/bar3d)
* [circle](https://www.programmingsimplified.com/c/graphics.h/circle)
* [cleardevice](https://www.programmingsimplified.com/c/graphics.h/cleardevice)
* [closegraph](https://www.programmingsimplified.com/c/graphics.h/closegraph)
* [drawpoly](https://www.programmingsimplified.com/c/graphics.h/drawpoly)
* [ellipse](https://www.programmingsimplified.com/c/graphics.h/ellipse)
* [fillellipse](https://www.programmingsimplified.com/c/graphics.h/fillellipse)
* [fillpoly](https://www.programmingsimplified.com/c/graphics.h/fillpoly)
* [floodfill](https://www.programmingsimplified.com/c/graphics.h/floodfill)
* [getarccords](https://www.programmingsimplified.com/c/graphics.h/getarccords)
* [getbkcolor](https://www.programmingsimplified.com/c/graphics.h/getbkcolor)
* [getcolor](https://www.programmingsimplified.com/c/graphics.h/getcolor)
* [getdrivername](https://www.programmingsimplified.com/c/graphics.h/getdrivername)
* [getimage](https://www.programmingsimplified.com/c/graphics.h/getimage)
* [getmaxcolor](https://www.programmingsimplified.com/c/graphics.h/getmaxcolor)
* [getmaxx](https://www.programmingsimplified.com/c/graphics.h/getmaxx)
* [getmaxy](https://www.programmingsimplified.com/c/graphics.h/getmaxy)
* [getpixel](https://www.programmingsimplified.com/c/graphics.h/getpixel)
* [getx](https://www.programmingsimplified.com/c/graphics.h/getx)
* [gety](https://www.programmingsimplified.com/c/graphics.h/gety)
* [graphdefaults](https://www.programmingsimplified.com/c/graphics.h/graphdefaults)
* [grapherrormsg](https://www.programmingsimplified.com/c/graphics.h/grapherrormsg)
* [imagesize](https://www.programmingsimplified.com/c/graphics.h/imagesize)
* [line](https://www.programmingsimplified.com/c/graphics.h/line)
* [lineto](https://www.programmingsimplified.com/c/graphics.h/lineto)
* [linerel](https://www.programmingsimplified.com/c/graphics.h/linerel)
* [moveto](https://www.programmingsimplified.com/c/graphics.h/moveto)
* [moverel](https://www.programmingsimplified.com/c/graphics.h/moverel)
* [outtext](https://www.programmingsimplified.com/c/graphics.h/outtext)
* [outtextxy](https://www.programmingsimplified.com/c/graphics.h/outtextxy)
* [pieslice](https://www.programmingsimplified.com/c/graphics.h/pieslice)
* [putimage](https://www.programmingsimplified.com/c/graphics.h/putimage)
* [putpixel](https://www.programmingsimplified.com/c/graphics.h/putpixel)
* [rectangle](https://www.programmingsimplified.com/c/graphics.h/rectangle)
* [sector](https://www.programmingsimplified.com/c/graphics.h/sector)
* [setbkcolor](https://www.programmingsimplified.com/c/graphics.h/setbkcolor)
* [setcolor](https://www.programmingsimplified.com/c/graphics.h/setcolor)
* [setfillstyle](https://www.programmingsimplified.com/c/graphics.h/setfillstyle)
* [setlinestyle](https://www.programmingsimplified.com/c/graphics.h/setlinestyle)
* [settextstyle](https://www.programmingsimplified.com/c/graphics.h/settextstyle)
* [setviewport](https://www.programmingsimplified.com/c/graphics.h/setviewport)
* [textheight](https://www.programmingsimplified.com/c/graphics.h/textheight)
* [textwidth](https://www.programmingsimplified.com/c/graphics.h/textwidth)