**Borland Graphics Interface (BGI) for Windows**

Version 6.0, August 9, 2004

The following functions are mostly from the original Borland Graphics Interface for DOS programs. The BGI graphics functions may also be used with Windows programs created by the Borland 5.0 compiler, the free GNU C++ compiler, and possibly other compilers. Extra Windows functions are also available, described in [www.cs.colorado.edu/~main/cs1300/doc/bgi/bgi.html](http://winbgim.codecutter.org/V6_0/doc/bgi.html). These extra functions are indicated below by [WIN]. Also, any of the functions that use colors can use [RGB colors](http://winbgim.codecutter.org/V6_0/doc/wincolor.html) in addition to the 16-color BGI palette.

**Functions:**

void [arc](http://winbgim.codecutter.org/V6_0/doc/arc.html) (int x, int y, int stangle, int endangle, int radius);

void [bar](http://winbgim.codecutter.org/V6_0/doc/bar.html) (int left, int top, int right, int bottom);

void [bar3d](http://winbgim.codecutter.org/V6_0/doc/bar3d.html) (int left, int top, int right, int bottom, int depth, int topflag);

ostringstream [bgiout](http://winbgim.codecutter.org/V6_0/doc/bgiout.html); [WIN

void [circle](http://winbgim.codecutter.org/V6_0/doc/circle.html) (int x, int y, int radius);

void [cleardevice](http://winbgim.codecutter.org/V6_0/doc/cleardevice.html) (void);

void [clearmouseclick](http://winbgim.codecutter.org/V6_0/doc/clearmouseclick.html)(int kind); [WIN

void [clearviewport](http://winbgim.codecutter.org/V6_0/doc/clearviewport.html) (void);

void [closegraph](http://winbgim.codecutter.org/V6_0/doc/closegraph.html) (int window=ALL\_WINDOWS); [WIN

int [converttorgb](http://winbgim.codecutter.org/V6_0/doc/wincolor.html) (int color); [WIN

void [delay](http://winbgim.codecutter.org/V6_0/doc/delay.html) (int millisec); [WIN

void [detectgraph](http://winbgim.codecutter.org/V6_0/doc/detectgraph.html) (int \*graphdriver, int \*graphmode);

void [drawpoly](http://winbgim.codecutter.org/V6_0/doc/drawpoly.html) (int numpoints, int \*polypoints);

void [ellipse](http://winbgim.codecutter.org/V6_0/doc/ellipse.html) (int x, int y, int stangle, int endangle, int xradius, int yradius);

void [fillellipse](http://winbgim.codecutter.org/V6_0/doc/fillellipse.html) (int x, int y, int xradius, int yradius);

void [fillpoly](http://winbgim.codecutter.org/V6_0/doc/fillpoly.html) (int numpoints, int \*polypoints);

void [floodfill](http://winbgim.codecutter.org/V6_0/doc/floodfill.html) (int x, int y, int border);

int [getactivepage](http://winbgim.codecutter.org/V6_0/doc/getactivepage.html) (void); [WIN

void [getarccoords](http://winbgim.codecutter.org/V6_0/doc/getarccoords.html) (struct arccoordstype \*arccoords);

void [getaspectratio](http://winbgim.codecutter.org/V6_0/doc/getaspectratio.html) (int \*xasp, int \*yasp);

int [getbkcolor](http://winbgim.codecutter.org/V6_0/doc/getbkcolor.html) (void);

int [getch](http://winbgim.codecutter.org/V6_0/doc/getch.html) (void); [WIN

int [getcolor](http://winbgim.codecutter.org/V6_0/doc/getcolor.html) (void);

int [getcurrentwindow](http://winbgim.codecutter.org/V6_0/doc/getcurrentwindow.html) (void); [WIN]

struct palettetype\* [getdefaultpalette](http://winbgim.codecutter.org/V6_0/doc/getdefaultpalette.html) (void);

int [getdisplaycolor](http://winbgim.codecutter.org/V6_0/doc/getdisplaycolor.html) (int color); [WIN

char\* [getdrivername](http://winbgim.codecutter.org/V6_0/doc/getdrivername.html) (void);

void [getfillpattern](http://winbgim.codecutter.org/V6_0/doc/getfillpattern.html) (char \*pattern);

void [getfillsettings](http://winbgim.codecutter.org/V6_0/doc/getfillsettings.html) (struct fillsettingstype \*fillinfo);

int [getgraphmode](http://winbgim.codecutter.org/V6_0/doc/getgraphmode.html) (void);

void [getimage](http://winbgim.codecutter.org/V6_0/doc/getimage.html) (int left, int top, int right, int bottom, void \*bitmap);

void [getlinesettings](http://winbgim.codecutter.org/V6_0/doc/getlinesettings.html) (struct linesettingstype \*lineinfo);

int [getmaxcolor](http://winbgim.codecutter.org/V6_0/doc/getmaxcolor.html) (void);

int [getmaxmode](http://winbgim.codecutter.org/V6_0/doc/getmaxmode.html) (void);

int [getmaxheight](http://winbgim.codecutter.org/V6_0/doc/getmaxheight.html) (void); [WIN

int [getmaxwidth](http://winbgim.codecutter.org/V6_0/doc/getmaxwidth.html) (void); [WIN

int [getmaxx](http://winbgim.codecutter.org/V6_0/doc/getmaxx.html) (void);

int [getmaxy](http://winbgim.codecutter.org/V6_0/doc/getmaxy.html) (void);

char\* [getmodename](http://winbgim.codecutter.org/V6_0/doc/getmodename.html) (int mode\_number);

void [getmoderange](http://winbgim.codecutter.org/V6_0/doc/getmoderange.html) (int graphdriver, int \*lomode, int \*himode);

void [getmouseclick](http://winbgim.codecutter.org/V6_0/doc/getmouseclick.html)(int kind, int& x, int& y); [WIN

void [getpalette](http://winbgim.codecutter.org/V6_0/doc/getpalette.html) (struct palettetype \*palette);

int [getpalettesize](http://winbgim.codecutter.org/V6_0/doc/getpalettesize.html) (void);

int [getpixel](http://winbgim.codecutter.org/V6_0/doc/getpixel.html) (int x, int y);

void [gettextsettings](http://winbgim.codecutter.org/V6_0/doc/gettextsettings.html) (struct textsettingstype \*texttypeinfo);

void [getviewsettings](http://winbgim.codecutter.org/V6_0/doc/getviewsettings.html) (struct viewporttype \*viewport);

int [getvisualpage](http://winbgim.codecutter.org/V6_0/doc/getvisualpage.html) (void); [WIN

int [getwindowheight](http://winbgim.codecutter.org/V6_0/doc/getwindowheight.html) (void); [WIN

int [getwindowwidth](http://winbgim.codecutter.org/V6_0/doc/getwindowwidth.html) (void); [WIN

int [getx](http://winbgim.codecutter.org/V6_0/doc/getx.html) (void);

int [gety](http://winbgim.codecutter.org/V6_0/doc/gety.html) (void);

void [graphdefaults](http://winbgim.codecutter.org/V6_0/doc/graphdefaults.html) (void);

char\* [grapherrormsg](http://winbgim.codecutter.org/V6_0/doc/grapherrormsg.html) (int errorcode);

int [graphresult](http://winbgim.codecutter.org/V6_0/doc/graphresult.html)(void);

unsigned [imagesize](http://winbgim.codecutter.org/V6_0/doc/imagesize.html) (int left, int top, int right, int bottom);

void [initgraph](http://winbgim.codecutter.org/V6_0/doc/initgraph.html) (int \*graphdriver, int \*graphmode, char \*pathtodriver);

int [initwindow](http://winbgim.codecutter.org/V6_0/doc/initwindow.html) (int width, int height, const char\* title="Windows BGI", int left=0, int top=0, bool dbflag=false, bool closeflag=true); [WIN]

int [installuserdriver](http://winbgim.codecutter.org/V6_0/doc/installuserdriver.html) (char \*name, int huge (\*detect)(void));

int [installuserfont](http://winbgim.codecutter.org/V6_0/doc/installuserfont.html) (char \*name);

bool [ismouseclick](http://winbgim.codecutter.org/V6_0/doc/ismouseclick.html)(int kind); [WIN

int [kbhit](http://winbgim.codecutter.org/V6_0/doc/kbhit.html) (void); [WIN

void [line](http://winbgim.codecutter.org/V6_0/doc/line.html) (int x1, int y1, int x2, int y2);

void [linerel](http://winbgim.codecutter.org/V6_0/doc/linerel.html) (int dx, int dy);

void [lineto](http://winbgim.codecutter.org/V6_0/doc/lineto.html) (int x, int y);

int [mousex](http://winbgim.codecutter.org/V6_0/doc/mousex.html) (void); [WIN

int [mousey](http://winbgim.codecutter.org/V6_0/doc/mousey.html) (void); [WIN

void [moverel](http://winbgim.codecutter.org/V6_0/doc/moverel.html) (int dx, int dy);

void [moveto](http://winbgim.codecutter.org/V6_0/doc/moveto.html) (int x, int y);

void [outtext](http://winbgim.codecutter.org/V6_0/doc/outtext.html) (char \*textstring);

void [outtextxy](http://winbgim.codecutter.org/V6_0/doc/outtextxy.html) (int x, int y, char \*textstring);

void [pieslice](http://winbgim.codecutter.org/V6_0/doc/pieslice.html) (int x, int y, int stangle, int endangle, int radius);

void [printimage](http://winbgim.codecutter.org/V6_0/doc/printimage.html) (

const char\* title=NULL, double width\_inches=7,

double border\_left\_inches=0.75, double border\_top\_inches=0.75,

int left=0, int right=0, int right=INT\_MAX, int bottom=INT\_MAX

); [WIN]

void [putimage](http://winbgim.codecutter.org/V6_0/doc/putimage.html) (int left, int top, void \*bitmap, int op);

void [putpixel](http://winbgim.codecutter.org/V6_0/doc/putpixel.html) (int x, int y, int color);

void [readimagefile](http://winbgim.codecutter.org/V6_0/doc/readimagefile.html) (

const char\* filename=NULL,

int left=0, int top=0, int right=INT\_MAX, int bottom=INT\_MAX

);

void [rectangle](http://winbgim.codecutter.org/V6_0/doc/rectangle.html) (int left, int top, int right, int bottom);

int [registerbgidriver](http://winbgim.codecutter.org/V6_0/doc/registerbgidriver.html) (void (\*driver)(void));

int [registerbgifont](http://winbgim.codecutter.org/V6_0/doc/registerbgifont.html) (void (\*font)(void));

void [registermousehandler](http://winbgim.codecutter.org/V6_0/doc/registermousehandler.html) (int kind, void h(int, int)); [WIN

void [restorecrtmode](http://winbgim.codecutter.org/V6_0/doc/restorecrtmode.html) (void);

[RGB functions:](http://winbgim.codecutter.org/V6_0/doc/wincolor.html) [WIN

COLOR(r,g,b),

RED\_VALUE(v), GREEN\_VALUE(v), BLUE\_VALUE(v),

IS\_BGI\_COLOR(v), IS\_RGB\_COLOR(v)

void [sector](http://winbgim.codecutter.org/V6_0/doc/sector.html) (int x, int y, int stangle, int endangle, int xradius, int yradius);

void [setactivepage](http://winbgim.codecutter.org/V6_0/doc/setactivepage.html) (int page);

void [setallpalette](http://winbgim.codecutter.org/V6_0/doc/setallpalette.html) (struct palettetype \*palette);

void [setaspectratio](http://winbgim.codecutter.org/V6_0/doc/setaspectratio.html) (int xasp, int yasp);

void [setbkcolor](http://winbgim.codecutter.org/V6_0/doc/setbkcolor.html) (int color);

void [setcolor](http://winbgim.codecutter.org/V6_0/doc/setcolor.html) (int color);

void [setcurrentwindow](http://winbgim.codecutter.org/V6_0/doc/setcurrentwindow.html) (int window); [WIN]

void [setmousequeuestatus](http://winbgim.codecutter.org/V6_0/doc/setmousequeuestatus.html)(int kind, bool status=true); [WIN

void [setfillpattern](http://winbgim.codecutter.org/V6_0/doc/setfillpattern.html) (char \*upattern, int color);

void [setfillstyle](http://winbgim.codecutter.org/V6_0/doc/setfillstyle.html) (int pattern, int color);

unsigned [setgraphbufsize](http://winbgim.codecutter.org/V6_0/doc/setgraphbufsize.html) (unsigned bufsize);

void [setgraphmode](http://winbgim.codecutter.org/V6_0/doc/setgraphmode.html) (int mode);

void [setlinestyle](http://winbgim.codecutter.org/V6_0/doc/setlinestyle.html) (int linestyle, unsigned upattern, int thickness);

void [setpalette](http://winbgim.codecutter.org/V6_0/doc/setpalette.html) (int colornum, int color);

void [setrgbpalette](http://winbgim.codecutter.org/V6_0/doc/setrgbpalette.html) (int colornum, int red, int green, int blue);

void [settextjustify](http://winbgim.codecutter.org/V6_0/doc/settextjustify.html) (int horiz, int vert);

void [settextstyle](http://winbgim.codecutter.org/V6_0/doc/settextstyle.html) (int font, int direction, int charsize);

void [setusercharsize](http://winbgim.codecutter.org/V6_0/doc/setusercharsize.html) (int multx, int divx, int multy, int divy);

void [setviewport](http://winbgim.codecutter.org/V6_0/doc/setviewport.html) (int left, int top, int right, int bottom, int clip);

void [setvisualpage](http://winbgim.codecutter.org/V6_0/doc/setvisualpage.html) (int page);

void [setwritemode](http://winbgim.codecutter.org/V6_0/doc/setwritemode.html) (int mode);

int [showerrorbox](http://winbgim.codecutter.org/V6_0/doc/showerrorbox.html) (const char \*message); [WIN

int [swapbuffers](http://winbgim.codecutter.org/V6_0/doc/swapbuffers.html) (void); [WIN]

int [textheight](http://winbgim.codecutter.org/V6_0/doc/textheight.html) (char \*textstring);

int [textwidth](http://winbgim.codecutter.org/V6_0/doc/textwidth.html) (char \*textstring);

void [writeimagefile](http://winbgim.codecutter.org/V6_0/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

); [WIN