

X-Window Graphics の基礎

X-Window のプログラムを簡単に作成するためにつくりました。以下の X_graphics.c と Xtest.c を同じディレクトリに入れ Xtest.c をコンパイルして実行してみてください。

Xtest.c を compile するときは ,

```
cc -L /usr/X11R6/lib/ -lX11 -lm Xtest.c
```

とします .

1. X-Graphics 関連の関数

- (a) X_open(400);
400 × 400 の X-Window が用意される . 左上の座標が (0,0) 右下の座標が (400,400) となる .
- (b) X_close();
X-Window を閉じる .
- (c) X_setcolor(n);
色を決める n : 0 ~ 15
- (d) X_putpixel(U, V);
点を打つ (U,V) : 座標
- (e) X_line(U1, V1, U2, V2);
線分を描く (U1,V1), (U2,V2) : 座標
- (f) X_rectangle(U, V, W, H);
長方形を描く (U,V) : 左上の座標 , W : 幅 , H : 高さ
- (g) X_circle(U, V, R);
円を描く (U,V) 中心: 半径 R の円
- (h) X_text(U, V, " ", n);
文を書く (U,V) : 左下の座標 , n : 文字数

2. 参考書

X-Window Ver.11 プログラミング (第 2 版) 日刊工業新聞社

X-Window 実用グラフィックス入門 日刊工業新聞社

```

/*****
/*      X_graphics.c      */
/*****

#include <X11/Xlib.h>
#include <X11/Xutil.h>

static Display      *disp;
static Window      root_win, win0, win1, win2;
static GC          gc;
static XSetWindowAttributes attr;
static unsigned long color[16];
static int         width=400;

/*****
/*  X_set_color();    16色の設定      */
/*****

void X_set_color16(void)
{
    static char colname[16][15] = {
        "black", "slate blue", "coral", "blue violet",
        "lime green", "spring green", "khaki", "light gray",
        "white", "blue", "red", "magenta",
        "green", "cyan", "yellow", "orange"};
    Colormap colormap;
    XColor c0, c1;
    int i;
    colormap = XDefaultColormap(disp, 0);
    for (i=0; i<16; i++){
        XAllocNamedColor(disp, colormap, colname[i], &c1, &c0);
        color[i]=c1.pixel;
    }
}

/*****
/*  X_open(w);    初期化  w:ウインドウの幅と高さ      */
/*****

```

```

void X_open(int w)
{
    disp=XOpenDisplay(NULL);
    root_win=XDefaultRootWindow(disp);
    X_set_color16();
    win0=XCreateSimpleWindow
        (disp,root_win,0,0,w+1,w+1,2,
         BlackPixel(disp,0),WhitePixel(disp,0));
    attr.backing_store=Always;
    XChangeWindowAttributes(disp,win0,CWBackingStore, &attr);
    XMapWindow (disp, win0);
    gc=XCreateGC (disp, root_win,0,0);
}

/*****
/*    X_close();    X-Windows 終了処理          */
*****/

void X_close(void)
{
    XFlush(disp);
    XFreeGC(disp,gc);
    XDestroyWindow(disp, win0 );
    XFlush(disp);
    XCloseDisplay(disp);
}

/*****
/*    X_setcolor(n);    表示色    n:色番号 (0-15)    */
*****/

void X_setcolor(int n)
{
    XSetForeground(disp,gc,color[n]);
}

/*****
/*    X_putpixel(U, V);    点を打つ    U,V : 座標    */
*****/

```

```

void X_putpixel(int U, int V)
{
    XDrawPoint(dis,win0,gc,U,V);
    XFlush(dis);
}

/*****
/*    X_line(U1, V1, U2, V2); 線分 U1,V1,U2,V2:座標 */
*****/

void X_line(int U1, int V1, int U2, int V2)
{
    XDrawLine(dis,win0,gc,U1, V1, U2, V2);
    XFlush(dis);
}

/*****
/*    X_rectangle(U, V, W, H);    矩形を描く    */
*****/

void X_rectangle(int U, int V, int W, int H)
{
    XDrawRectangle(dis,win0,gc,U, V, W, H);
    XFlush(dis);
}

/*****
/*    X_fill_rectangle(U, V, W, H);    矩形の中を塗る    */
*****/

void X_fill_rectangle(int U, int V, int W, int H)
{
    XFillRectangle(dis,win0,gc,U, V, W, H);
    XFlush(dis);
}

/*****
/*    X_circle(U, V, W);    円を描く    */
*****/

```

```

/*****/

void X_circle(int U, int V, int R)
{
    XDrawArc(disp,win0,gc,U-R, V-R, 2*R, 2*R, 0*64, 360*64);
    XFlush(disp);
}

/*****/
/*    X_fill_circle(U, V, W);    円の中を塗る    */
/*****/

void X_fill_circle(int U, int V, int R)
{
    XFillArc(disp,win0,gc,U-R, V-R, 2*R, 2*R,0*64,360*64);
    XFlush(disp);
}

/*****/
/*    X_text(U, V, "    ", n);    文を書く    */
/*****/

void X_text(int U, int V, char *s, int n)
{
    XDrawString(disp, win0, gc, U, V, s, n);
    XFlush(disp);
}

```

```

/*****
/*      X_test.c      */
*****/

#include <stdio.h>
#include "X_graphics.c"

main()
{
    int i;

    X_open(400);

    for(i=0;i<16;i++){
        X_setcolor(i);
        X_line(10,10+i,200,10+i);
    }
    X_setcolor(0);
    for(i=1;i<5;i++){
        X_putpixel(40+5*i,40);
    }
    X_setcolor(3);
    X_rectangle(100,100,200,200);
    X_setcolor(2);
    X_circle(100,100,100);
    X_setcolor(1);
    X_text(110,250,"X-Window",8);

    getchar();
    X_close();
}

```