

```
// generated by Fast Light User Interface Designer (fluid) version 1.0302

#include "lab11.h"
/**
```

1 Design

1. Subclass one of the existing `Fl_Widget` classes (either `Fl_Window` or `Fl_Box`) and implement our own version of `draw()`.
2. Create an instance (object or variable) of our new class in `main`.
3. In the overridden `draw` function, put the code to draw the graph.

```
    plot data as we read it from the file in a loop
    Draw the dots (astrixs)
    we can plot the dots using
    (fl_point(int x, int y))
    and connect each to its previous dot using line.
    imoreder to draw the line we can use
    (fl_rect(int x, int y, int w, int h) )
```

4. Create the call back for the animation and use (`Fl::add_timeout`) to kick it off. Code the cycle of the image.

```

*/
//
#include <FL/Fl_PNG_Image.H>
static Fl_Box* bug_box[2];
static Fl_PNG_Image* bug_images[2];
static Fl_Group* bug;

bool sky_ended() {
    return sky->x() + sky->w() < win->x() + win->w();
}

void cb_sky(void*) {
    bug->position(bug->x()+(rand()%100),bug->y()-(rand()%100));
    if((bug->x()+bug->w())>(win->x()+win->w()) || (bug->y()<0)) {
        bug->position(win->w()/2,win->h()/2);
        printf("Reached!");
    }
    win->redraw();
    Fl::repeat_timeout(0.05,cb_sky);
}

void load_images() {
    bug = new Fl_Group(0,400,55,24);
    for (int i = 0; i < 2; i++)
    {
        bug_box[i] = new Fl_Box(0,400,55,24+(i*50));
        bug_images[i] = new Fl_PNG_Image("ball.png");
        bug_box[i]->image(bug_images[i]);
        bug->add(bug_box[i]);
    }
}

Fl_Double_Window *win=(Fl_Double_Window *)0;

Fl_Box *sky=(Fl_Box *)0;

int main(int argc, char **argv) {
    { win = new Fl_Double_Window(845, 698);

```

```

    { Fl_Box* o = sky = new Fl_Box(0, -40, 1280, 1024);
      Fl_JPEG_Image* sky_image = new Fl_JPEG_Image("sky.jpg");
      o->image(sky_image);
    } // Fl_Box* sky
    win->end();
} // Fl_Double_Window* win
load_images();
win->add(bug);
Fl::add_timeout(0,cb_sky);
win->show(argc, argv);
return Fl::run();
}

```