Discussion 5

In the header file

- #ifndef _graphics_h
- #define _graphics_h
- void funcs();
- ……
- #endif

Why do we need these preprocessor lines?

Later, when we use #include “graphics.h”

Goodness

- Building time
- Errors related to multiple inclusion

Compiling(1)

- Preprocessing: gcc -E test.c -o test.i
- Compiling and optimization: gcc -S test.i -o test.s
- Assembling: gcc -c test.s -o test.o
- Linking: gcc test.o -o test
- Get the executable file: test
- All-in-one: gcc test.c -o test

Compiling(2)

- You can ignore the .i files and .s files
- Pay attention to the binary file: test.o
- Link multiple binary files: gcc -o test test1.o test2.o test3.o
- Later this quarter you will learn how to write Makefile
- Intermediate files:
  - single .c file
  - split into two files

with header file

- With only multiple source .c files:
  - gcc -o test test1.c test2.c test3.c
- With multiple files include header files:
  - Should we compile with the header file?
    - $gcc -o test test.h test.c main.c ?
    - or $gcc -o test test.c main.c ?

- Function definition in Header file?
  - Can we do this? Yes, we can.
    - Should we do this??? No, we are not encouraged to do this.
- Force warning during compiling:
  - gcc -Wall test.c -o test

Remote Graphics

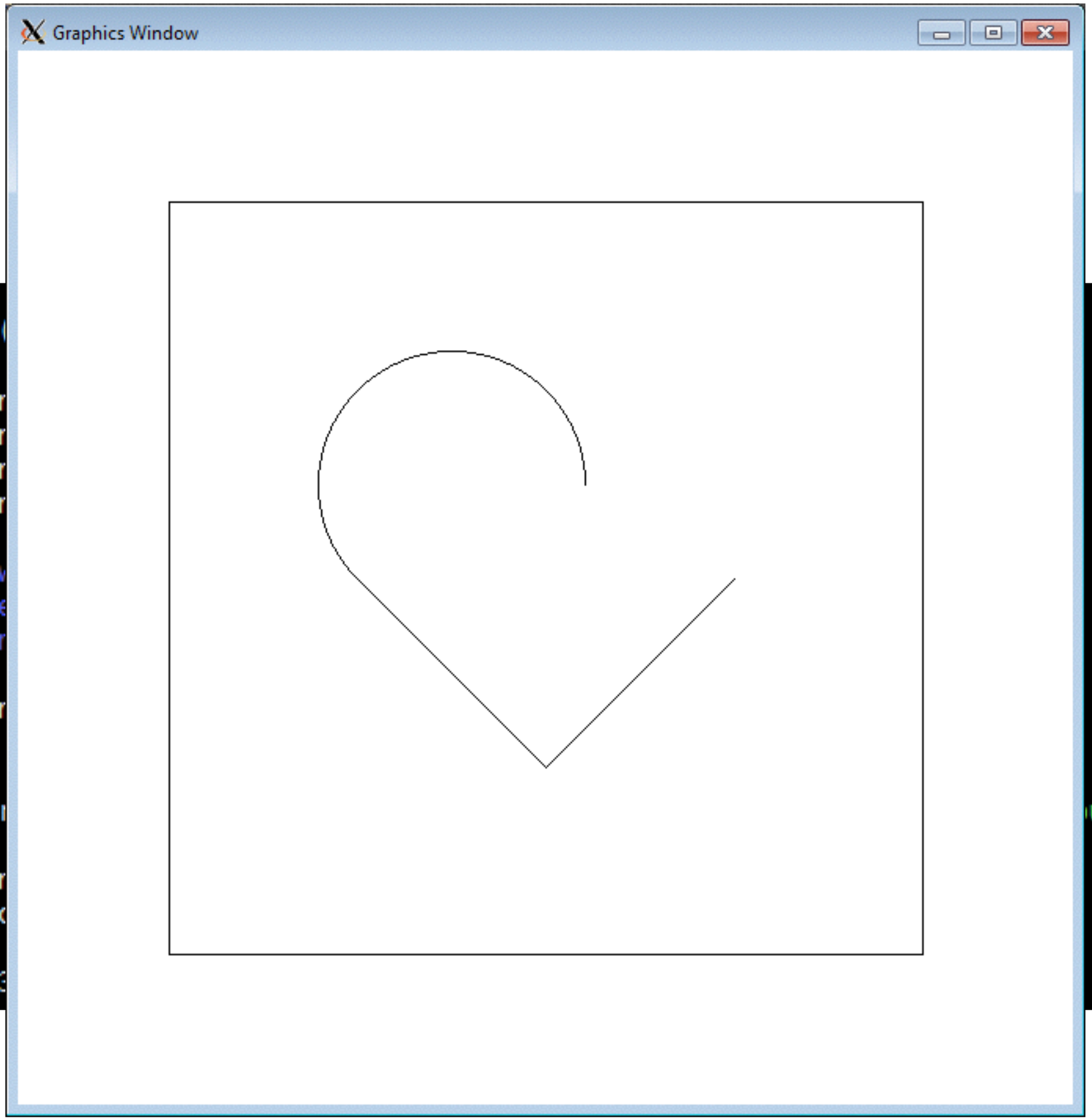
- On Windows just use MobaXterm.
- On Mac, you need to install the XQuartz
  - http://xquartz.macosforge.org/landing/
- use “ssh -X usrname@pc42.cs.ucdavis.edu”:
  - This symbol will appear in the Dock: 

```
void DrawHeart(
{
        MovePer
        DrawLir
        MovePer
        DrawLir

        // Draw
        // Move
//      MovePer
        double
        DrawCer
}

void DrawCenter                                          uble end)
{
        MovePer
        DrawArc
}
"graphics.c" 53
```

Graphics Window

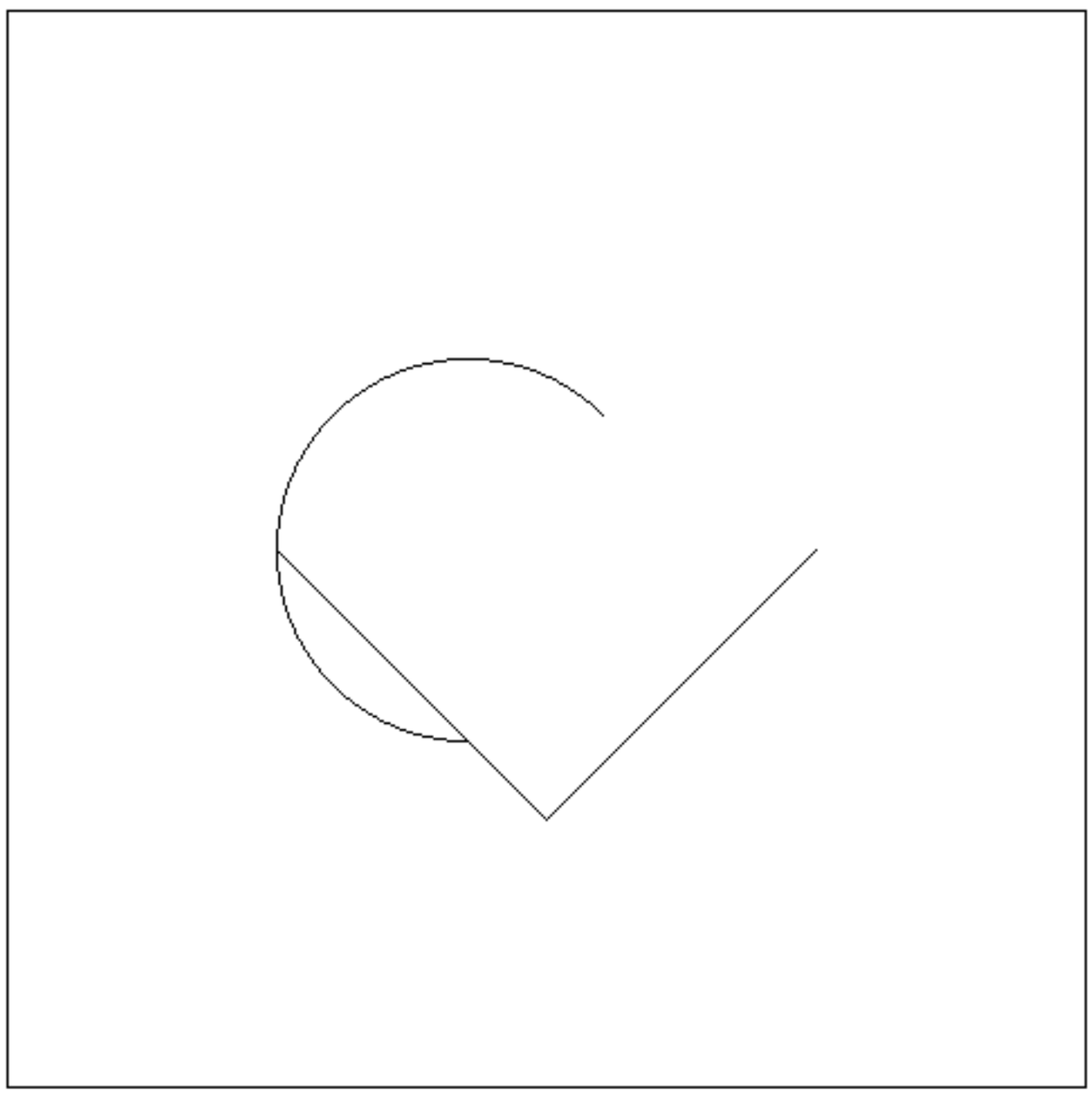


```
void DrawHeart
{
        MovePe
        DrawLi
        MovePe
        DrawLi

        // Dra
        // Mov
//      MovePe
        double
        DrawCe
}

void DrawCente                                                    louble end)
{
        MovePe
        DrawAr
}
```

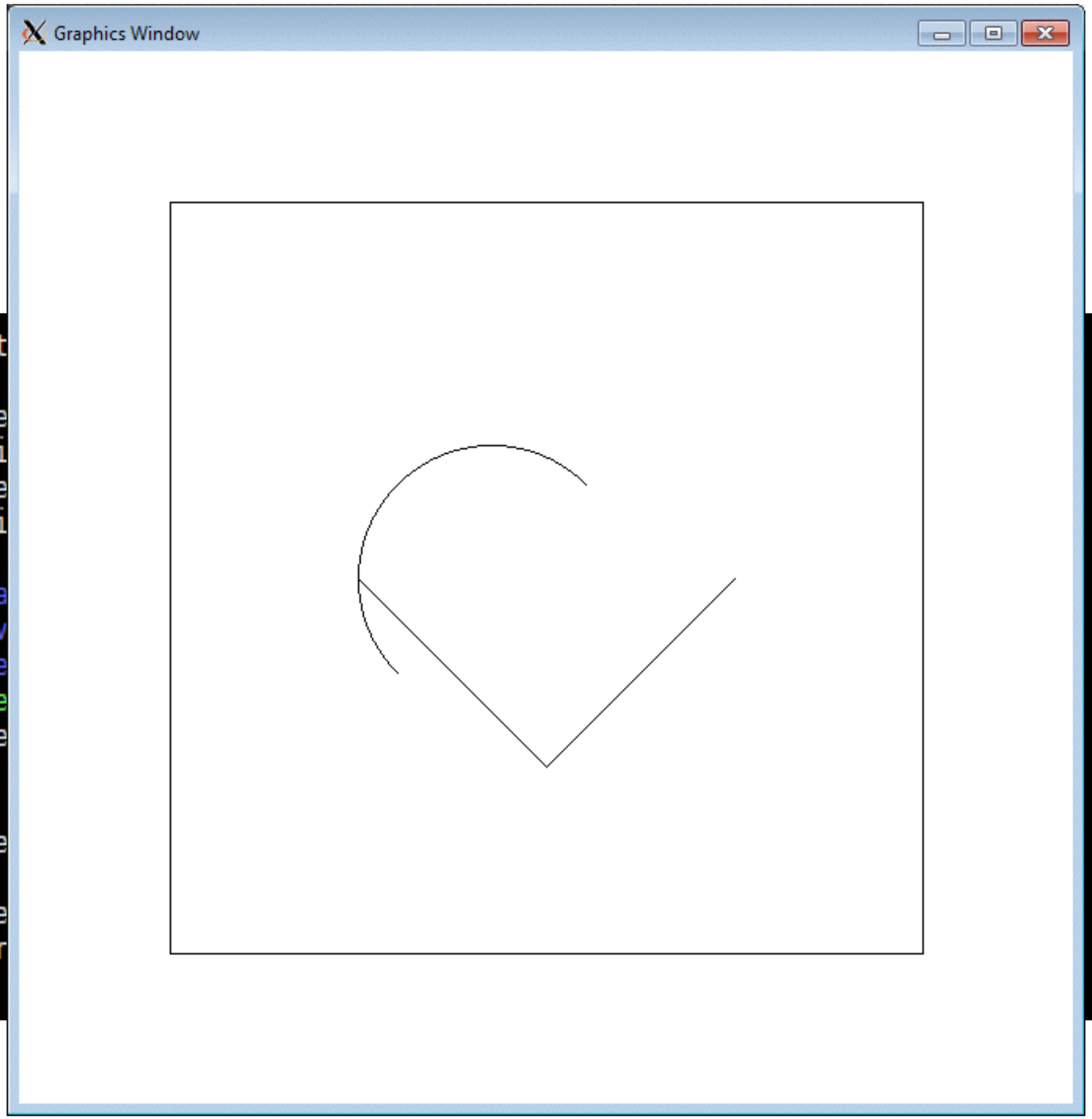
Graphics Window

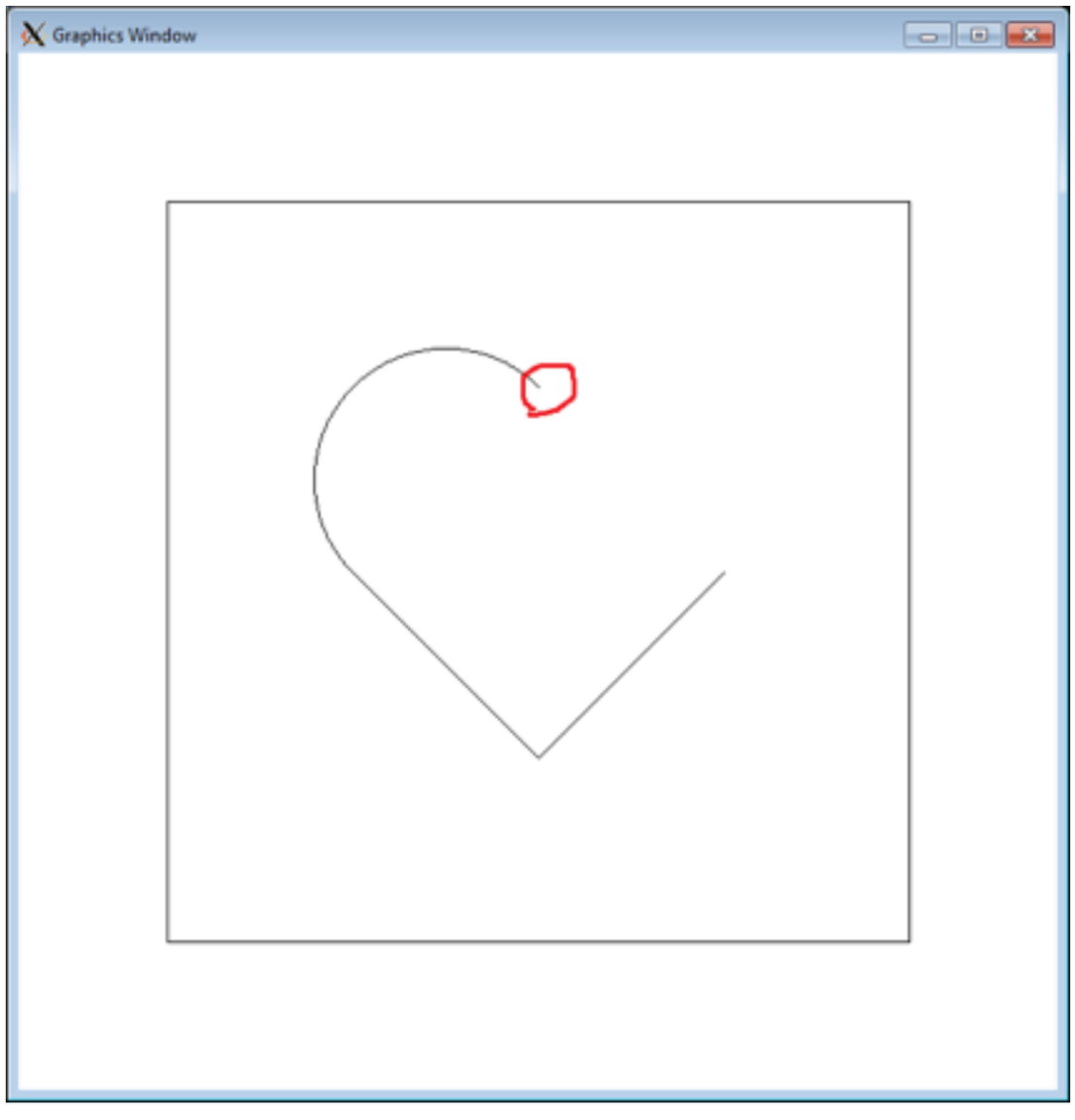


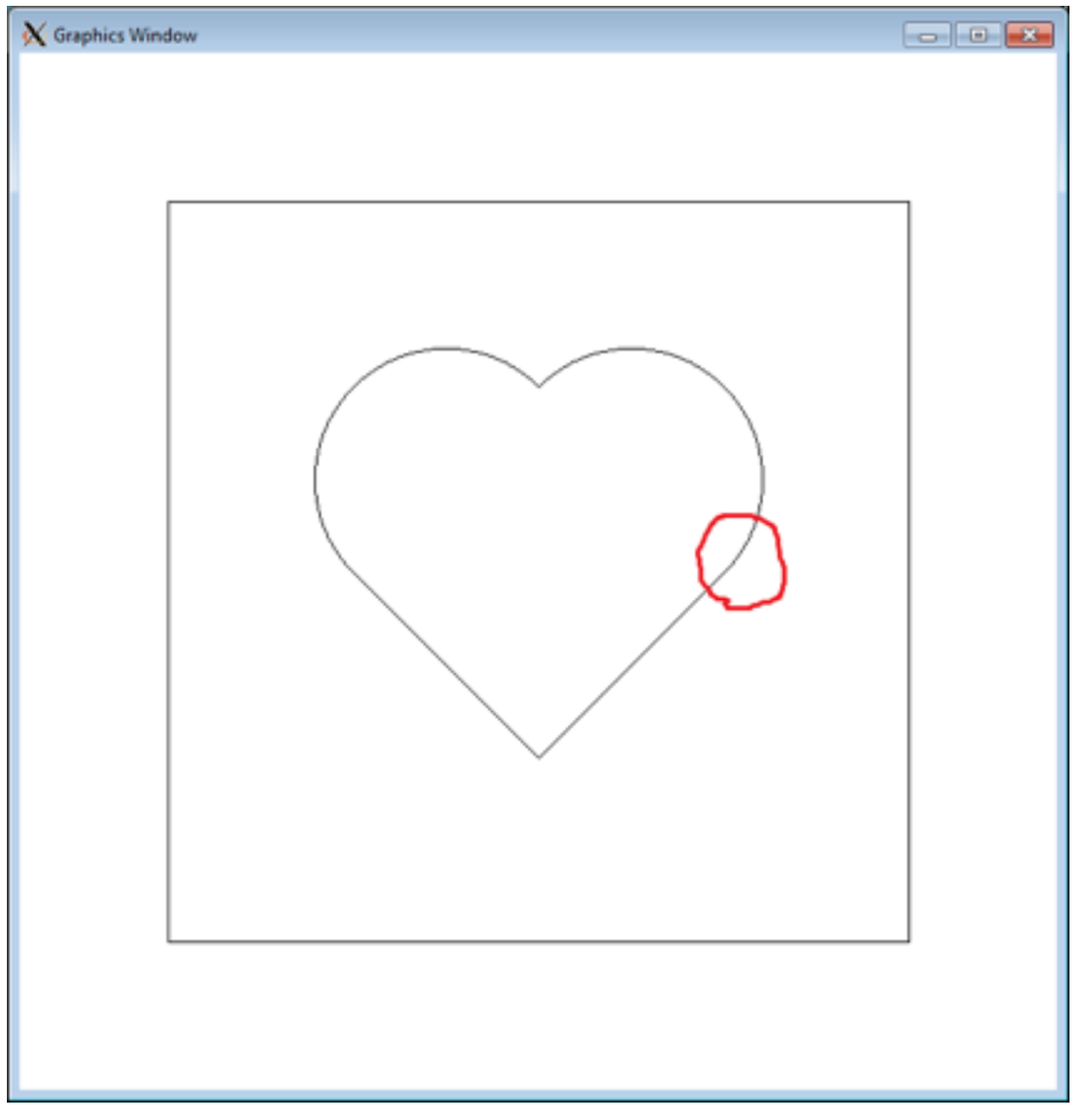
```
void DrawHeart
{
        MovePe
        DrawLi
        MovePe
        DrawLi

        // Dra
        // Mov
//      MovePe
        double
        DrawCe
}

void DrawCente                                          double end)
{
        MovePe
        DrawAr
}
```




Graphics Window


Graphics Window