

《程序设计专题》实验报告 3

图形程序设计

1. 实验目的

通过实验：

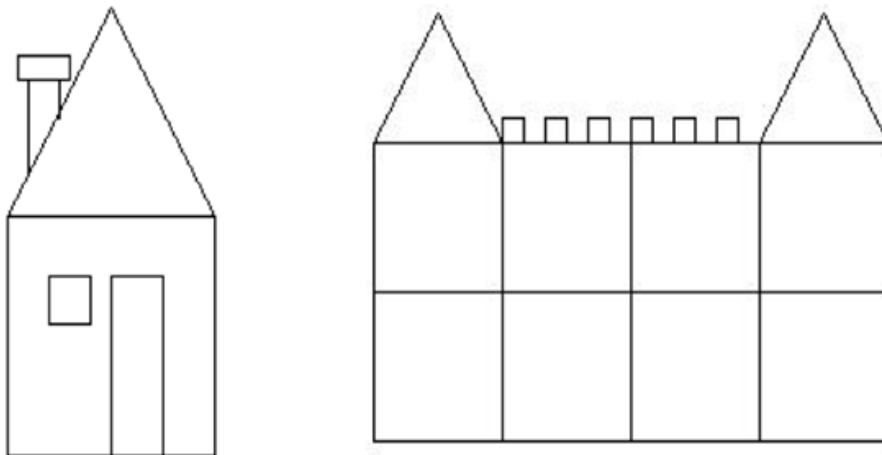
- 1) 掌握图形程序设计的基本概念
- 2) 掌握第三方库 `libgraphics` 的使用
- 3) 掌握基本的工程管理方法

2. 实验内容

- 1) 完成下面的几个绘图实验。
- 2) 打包各实验的工程文件，压缩为.zip 格式，上传至学在浙大，不需要写实验报告。上传的压缩文件命名为学号_姓名_HW3.zip。注意：各个项目请清除掉所有的.o 文件以及.exe 文件，评分时会重新编译各个项目。可以参考 PPT 以及 tutorial 中的工程组织方法，确保上传的各个项目可以不用重新配置环境直接编译运行。

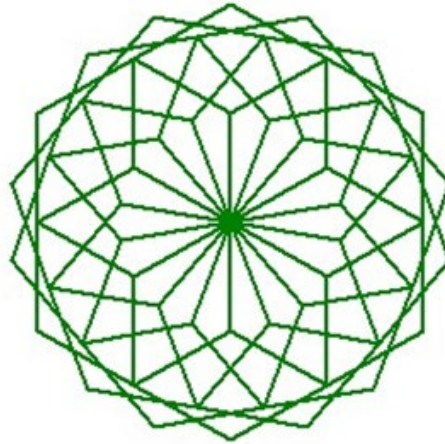
3. 实验题目

- 1) 画房子，写程序画出下面的房子之一。



- 2) 在现有的 `libgraphics` 库的基础上，设计一个新图形绘制系统。除了已有的笔的概念（有当前位置）之外，再增加一个表示当前角度的全局变量，初始值为 0，表示坐标轴的横轴方向（正东）。然后实现以下两个函数：

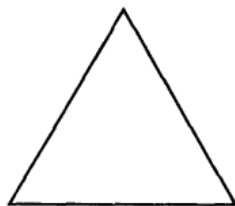
void forward(double distance); /*沿当前方向画线，长度为
distance，当 distance 为负数时后退*/
void turn (double angle); /*顺时针旋转角度 angle，角度单位为
DEG，当 angle 为负数时逆时针旋转*/
void move(double distance); /*沿当前方向移动笔（不画线），长度
为 distance，当 distance 为负数时后退*/
在实现以上全局变量和函数的基础上，用以上函数画出下面的图形：



- 3) 做一个随笔画程序，当鼠标左键按下拖动时，在窗口中随着鼠标位置画出轨迹，当鼠标左键抬起时则不画。
- 4) 做一个在图形窗口的输入程序。当用户按键时，在图形窗口出现按下的键所代表的字符，并像图形界面的输入框一样自动向右递进，按下回车键结束输入，将用户输入的内容在终端窗口显示出来。并在此基础上，实现当前位置光标闪烁，并支持左右方向键和两个删除键。
- 5) 绘制 Koch 雪花（结合递归）（以下仅帮助理解 Koch 雪花）

5. 20世纪70年代后期，IBM的一个研究员Benoit Mandelbrot出版了一本书，这本书引起了不小的轰动。这本书的主题是关于一个有趣的数学现象，叫分形（fractal），它是一种几何结构，该几何结构由一个形状以不同规模多次递归复制组成。数学家们早就发现了分形，但是到20世纪80年代才出现了一股关注分形的热潮。部分原因是计算机的发展使人们可以更深入、更多地研究分形。

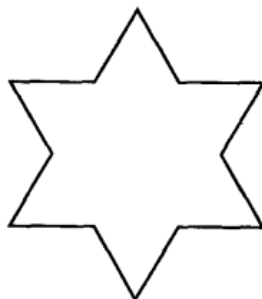
最早的一种分形之一是*koch*雪花，这是以它的发明者Helge von koch命名。*Koch*雪花以一个等边三角形开始：



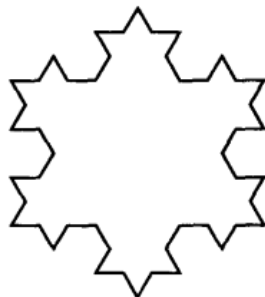
它被称为0阶*koch*分形。之后这个图形被逐步修改生成了更高阶的分形。在每一步修改中，三角形的每一条边被分为三段，中间一段线段被一个三角形的凸形取代。这样，如果你把三角形的每一条边都用下图做改变：



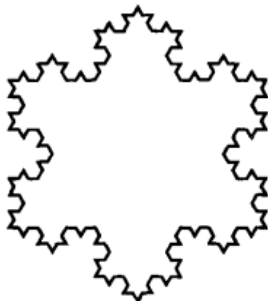
你将得到1阶*koch*分形：



如果你把这个图形中的每一条边再用一条有凸起三角形的线段取代，你将得到2阶*koch*分形：



再次用同样的方法替换图中的每一条边，将得到下图所示的3阶*koch*分形，看起来像一片雪花：



写一个递归函数，画出koch分形雪花，使用第7章的图形库在屏幕上显示这个图形。你的主程序应该读入你想生成的分形的阶数，以及原0阶三角形的边长。

在写这个程序时，你最好定义一个过程DrawPolarLine，让它画一条长为 r ，角度为 θ 的边，该角度是从 x 轴逆时针方向转过的度数，如Draw Arc的角度那样，实现如下：

```
static void DrawPolarLine(double r, double theta)
{
    double radians;

    radians = theta / 180 * 3.1415926535;
    DrawLine(r * cos(radians), r * sin(radians));
}
```

DrawPolarLine这个名字来自数学概念中的极坐标 (polar coordinate)，极坐标由半径 r 和角 θ 确定。