**模块0 绘制图案**

**一 实验目的**

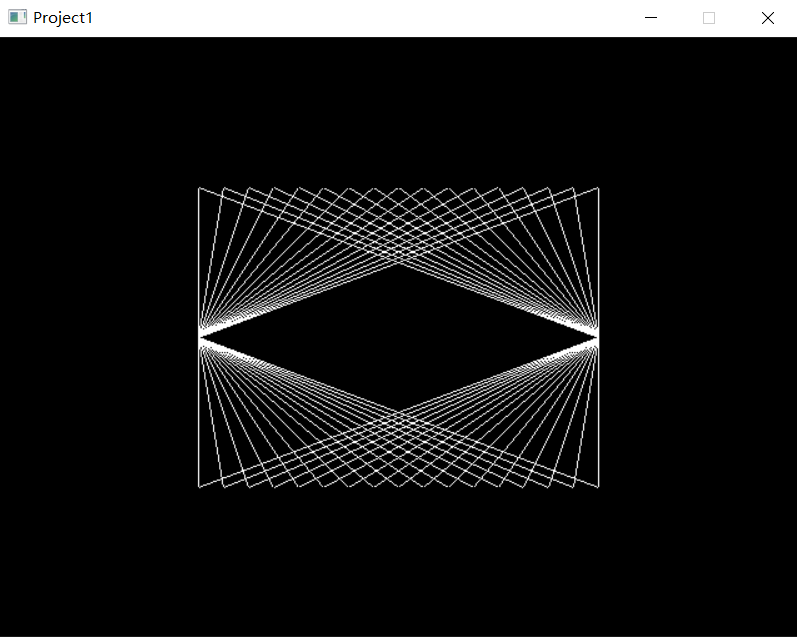
1. 熟悉相关的图形软件的运行环境，掌握图形函数绘制图案的方法，包括图形初始化，画点、线、圆、填充等常用绘图函数

**二 实验要求**

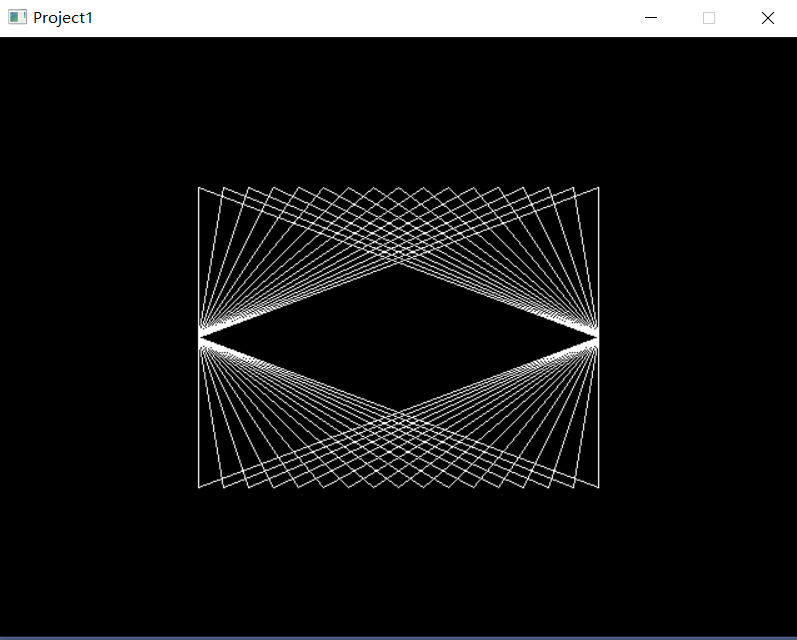
1. 能利用图形函数，正确调试运行程序
2. 熟悉网站（<http://www.easyx.cn/>）的使用和相关内容的阅读

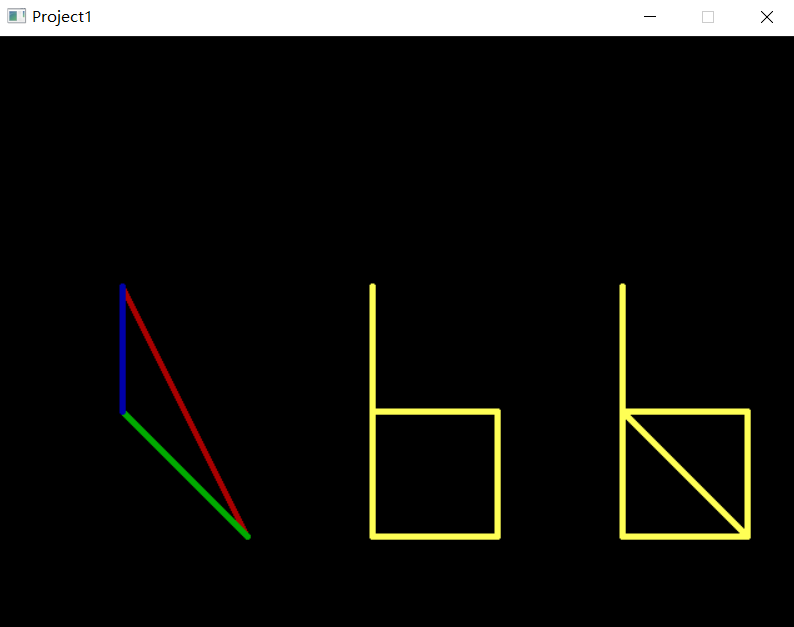
**三 实验内容**

0：样例——多笔绘制如图所示的图形

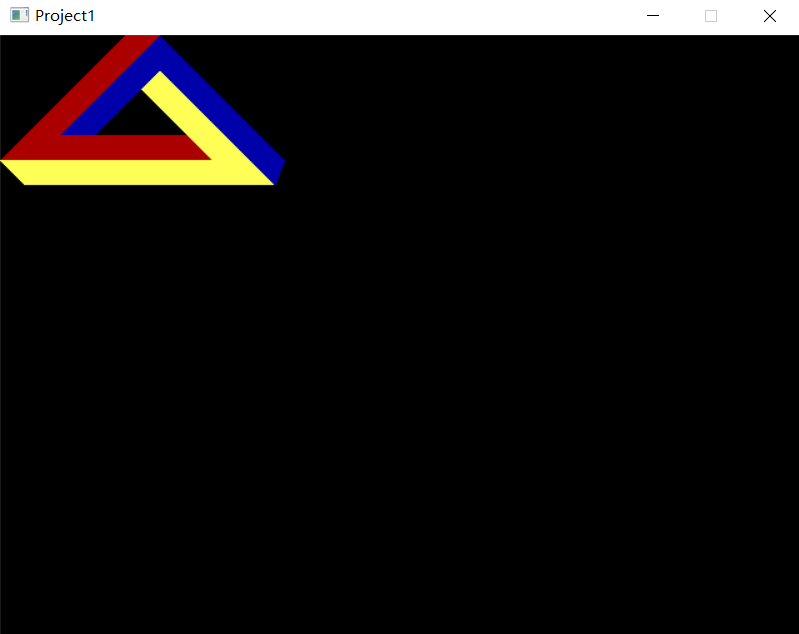
**画图思路：**通过for循环寻找初始点上方和下方两排点阵列的坐标，分别将初始点和for循环当前确认的上点和下点进行画线，从左向右依次完成。实验结果如下图所示：

1：一笔绘制如图所示的图形（lineto函数）

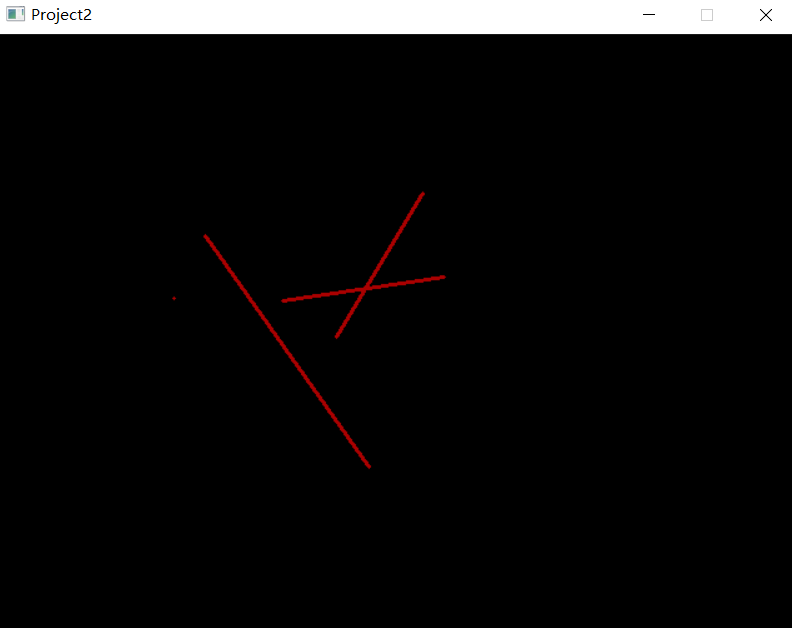
**画图思路：**通过for循环寻找初始点及其上方和下方组合后能够构成平行四边形的点，一次循环中画一个平行四边形，上方从左向右依次完成，下方从右向左依次完成。实验结果如下图所示：

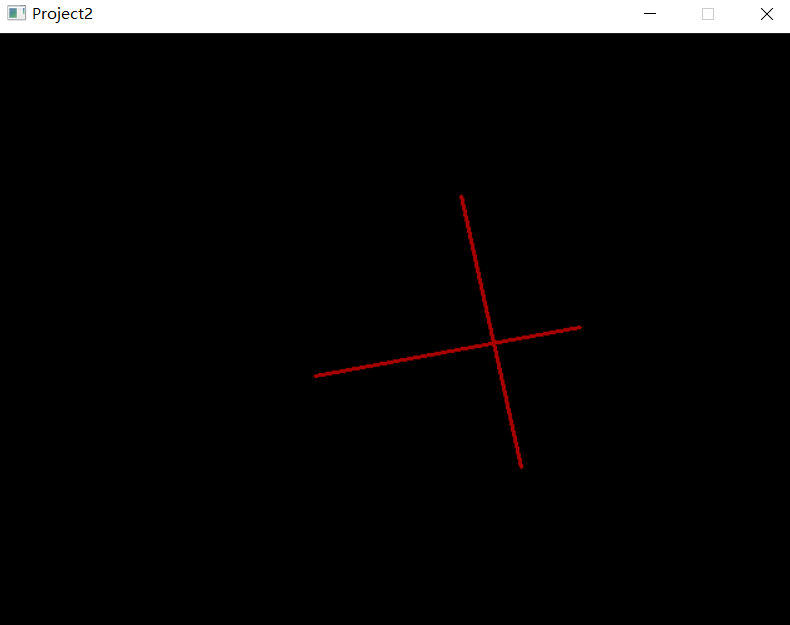
**画图思路：**通过分析需要绘制的三个图像，确定每个转折点的坐标，然后在每个图像中寻找一条欧拉通路，通过moveto函数设置一个可行的欧拉通路起点，再通过lineto函数实现通路。实验结果如下图所示：

2：绘制如图所示三角形并填充

**画图思路：**首先使用setfillcolor函数设置内部的填充颜色，再通过确定每个颜色所在的七边形的数组坐标，使用solidpolygon函数构建填充图形。实验结果如下图所示：

3：模仿鼠标操作题（示例程序/鼠标操作/cpp源文件），实现鼠标画线的功能

**画图思路1：**定义消息变量并获取鼠标和按键的消息，通过switch-case函数判定鼠标的操作，点击左键时调用moveto函数，点击右键时设置线条的样式和颜色并调用lineto函数。实验结果如下图所示：

**画图思路2：**定义消息变量并获取鼠标和按键的消息，通过switch-case函数判定鼠标的操作，按下左键时调用moveto函数，松开左键时设置线条的样式和颜色并调用lineto函数。实验结果如下图所示：

**四 程序说明**

Project中程序的调用：

当前.cpp文件的属性——常规——从生成中排除中选择否，其他文件选择是

|  |
| --- |
| 0题 |
| //////////////////////////////////////////////////////  // 程序名称：多笔绘制  // 功 能：多笔绘制如图所示的图形  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h> // 引用图形库头文件  #include <conio.h>  int main()  {  initgraph(640, 480); // 创建绘图窗口，大小为 640x480 像素  for (int x = 160; x <= 480; x += 20) {  line(160, 240, x, 120);  line(160, 240, x, 360);  line(480, 240, 640 - x, 120);  line(480, 240, 640 - x, 360);  }  \_getch(); // 按任意键继续  closegraph(); // 关闭绘图窗口  return 0;  } |
| 1题：实现1 |
| //////////////////////////////////////////////////////  // 程序名称：一笔绘制  // 功 能：一笔绘制0题的图像  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h> // 引用图形库头文件  #include <conio.h>  int main()  {  initgraph(640, 480); // 创建绘图窗口，大小为 640x480 像素  moveto(160, 240);  for (int x = 160; x <= 480; x += 20) {  lineto(x, 120);  lineto(480, 240);  lineto(480 - (x - 160), 360);  lineto(160, 240);  }  \_getch(); // 按任意键继续  closegraph(); // 关闭绘图窗口  return 0;  } |
| 1题：实现2 |
| //////////////////////////////////////////////////////  // 程序名称：一笔绘制  // 功 能：一笔绘制三角形、b、b中一杠的图像  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h> // 引用图形库头文件  #include <conio.h>  #include <math.h>  int main()  {  initgraph(640, 480); // 创建绘图窗口，大小为 640x480 像素  //三角形  moveto(100, 200);  setlinestyle(PS\_SOLID | PS\_JOIN\_BEVEL, 5);  setlinecolor(RED);  lineto(200, 400);  setlinecolor(GREEN);  lineto(100, 300);  setlinecolor(BLUE);  lineto(100, 200);  //b  moveto(300, 200);  setlinestyle(PS\_SOLID | PS\_JOIN\_BEVEL, 5);  setlinecolor(YELLOW);  lineto(300, 400);  lineto(400, 400);  lineto(400, 300);  lineto(300, 300);  //b加一个杠  moveto(500, 200);  setlinestyle(PS\_SOLID | PS\_JOIN\_BEVEL, 5);  setlinecolor(YELLOW);  lineto(500, 400);  lineto(600, 400);  lineto(600, 300);  lineto(500, 300);  lineto(600, 400);  \_getch(); // 按任意键继续  closegraph(); // 关闭绘图窗口  return 0;  } |
| 2题 |
| //////////////////////////////////////////////////////  // 程序名称：填充绘制  // 功 能：绘制如图所示三角形并填充  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h> // 引用图形库头文件  #include <conio.h>  int main()  {  initgraph(640, 480); // 创建绘图窗口，大小为 640x480 像素  int point1[] = { 100,0,128,0,48,80,150,80,170,100,0,100,100,0 };  int point2[] = { 0,100,20,120,220,120,128,28,113,43,170,100,0,100 };  int point3[] = { 76,80,128,28,220,120,228,100,128,0,48,80,76,80 };  /\*  moveto(100, 0);  for (int i = 2; i < 14; i+=2) {  lineto(point1[i], point1[i + 1]);  }  \*/  setfillcolor(RED);  solidpolygon((POINT\*)point1, 7);  setfillcolor(YELLOW);  solidpolygon((POINT\*)point2, 7);  setfillcolor(BLUE);  solidpolygon((POINT\*)point3, 7);  \_getch(); // 按任意键继续  closegraph(); // 关闭绘图窗口  return 0;  } |
| 3题：实现1 |
| //////////////////////////////////////////////////////  // 程序名称：模仿鼠标操作  // 功 能：点击鼠标左键确定线段起点，点击鼠标右键确定终点并画图  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h>  int main()  {  // 初始化图形窗口  initgraph(640, 480);  ExMessage m; // 定义消息变量  while (true)  {  // 获取一条鼠标或按键消息  m = getmessage(EX\_MOUSE | EX\_KEY);  switch (m.message) {  case WM\_LBUTTONDOWN:  moveto(m.x, m.y);//确定起点  case WM\_RBUTTONDOWN:  setlinestyle(PS\_SOLID | PS\_JOIN\_BEVEL, 3);//设置style  setlinecolor(RED);//设置color  lineto(m.x, m.y);//确定重点并连线  case WM\_KEYDOWN:  if (m.vkcode == VK\_ESCAPE)  return 0; // 按 ESC 键退出程序  }  }  // 关闭图形窗口  closegraph();  return 0;  } |
| 3题：实现2 |
| //////////////////////////////////////////////////////  // 程序名称：模仿鼠标操作  // 功 能：按下鼠标左键确定线段起点，抬起左键确定终点并画图  // 编译环境：VS2019，EasyX\_20220116  // 作 者：夏婉可<2020301010225><1597493790@qq.com>  // 最后修改：2022-2-24  #include <graphics.h>  int main()  {  // 初始化图形窗口  initgraph(640, 480);  ExMessage m; // 定义消息变量  while (true)  {  // 获取一条鼠标或按键消息  m = getmessage(EX\_MOUSE | EX\_KEY);  switch (m.message) {  case WM\_LBUTTONDOWN:  moveto(m.x, m.y);//确定起点  case WM\_LBUTTONUP:  setlinestyle(PS\_SOLID | PS\_JOIN\_BEVEL, 3);//设置style  setlinecolor(RED);//设置color  lineto(m.x, m.y);//确定重点并连线  case WM\_KEYDOWN:  if (m.vkcode == VK\_ESCAPE)  return 0; // 按 ESC 键退出程序  }  }  // 关闭图形窗口  closegraph();  return 0;  } |