

# OOP Final Project 1 Documentation

## 说明

本项目为张洵恺、朱俸民、李思涵小组在2013-2014学年春季学期的面向对象程序设计课程的期末项目之一。本项目的实现者为张洵恺。

本项目参考了一些论文、网络博文和算法书籍，但所有代码均为原创。

本项目在Github上的repo地址为 <https://github.com/badpoet/EMST> .

## 编译

### 环境准备

出于跨平台性的考虑，本项目使用CMake工具构建，因此，在编译此程序之前，请先安装CMake工具。

另外，本项目的一部分使用了OpenCV开源代码库，如果想编译该部分，需要安装OpenCV，并使其与CMake关联。

如果用户使用apt(on Ubuntu)或brew(on Mac OS X)安装CMake和OpenCV，关联是由其自动解决的。推荐用户使用这种方式。

### 编译

在此以\*nix系统上的操作为例。

首先拷贝或下载源码。如果需要下载，请使用：

```
git clone "https://github.com/badpoet/EMST"
```

进入EMST文件夹，建立构建目录：

```
mkdir build
```

使用CMake构建：

```
cmake ..  
make
```

项目将在bin目录下生成三个可执行文件：bin/generator，bin/emst，bin/viewer.

### 可选的编译

在三个可执行文件中，viewer是可视化EMST运算结果的工具，并不是程序所必需的。如果用户没有安装OpenCV库，viewer将不能编译。此时可以通过修改CMakeLists.txt不编译此文件。首先回到EMST目录下，再编辑src\CMakeLists.txt文件。

将其中第13行注释掉，也即变成：

```
# add_subdirectory(viewer)
```

保存退出后，再按照之前所描述的方式构建，即可生成全部代码。

## 使用

### generator

generator是一个基于随机的测试用例生成器。

generator接受两个参数，分别为点的个数、分布正方形的半边长。使用方法如下：

```
bin/generator 500 1000
```

这样就可以生成500个x, y坐标分布在[-1000, 1000]的二维点，并将结果输出到graph.in文件中。

### emst

emst是计算欧几里得最小生成树的主程序，接受一个参数为点分布的描述文件，例如generator生成的graph.in。

```
bin/emst graph.in
```

得到的屏幕输出为

```
EMST baseline:
14735.6
EMST answer:
14735.6
```

其中baseline是用朴素的Prim算法计算出的结果，answer是论文实现的结果。在我的多次测试中，当节点总数不太小时，二者的值均相等。

同时，还会生成emst.output.out文件和graph.output.out文件。后者是解决ENP问题之后所生成的图，而前者是该图上的最小生成树。

### viewer

viewer可以将上文所说的.out文件可视化，是使用OpenCV绘图的、大小自适应的工具。接受一个.out文件作为参数。使用方法为：

```
bin/viewer emst.output.out
```

在屏幕显示的同时，也会生成一个JPEG文件，供用户检查对比。

## 源码简介

include/ 头文件。

src/generator/ generator的源码。

src/viewer/ viewer的源码。

src/mst/ 最小生成树算法的实现。目前仅实现了prim算法，只需继承MSTSolver类就可以自己实现想要的算法。

src/enp/ 八个最邻近点的算法的实现。目前使用了 $O(n^2)$ 的朴素算法。只需继承ENPSolver类就可以实现效率更高的算法（如姚期智教授论文中的算法）。

总体来看，算法的复杂度为 $O(n^2)$ 。但实际上，在运行中可以明显发现，实现的基于ENP算法的方法要大大快于作为baseline的Prim算法。

## 代码亮点

- 跨平台的CMake构建工具；
- 基于OpenCV的可视化工具；
- 结合Strategy设计模式的整体设计。