

# README

## 项目介绍:

尊敬的杜老师，助教老师：

这次大作业我选择的是《**最小生成树的可视化形成**》这一选题，代码主体基于Python。

其中利用了Tkinter库作为可视化界面的制作，这是一款Tkinter是Python的标准GUI库。Python使用Tkinter可以快速的创建GUI应用程序。由于Tkinter是内置到python的安装包中、只要安装好Python之后就能import Tkinter库、而且IDLE也是用Tkinter编写而成、对于简单的图形界面Tkinter还是能应付自如。

很幸运之前对深度学习很感兴趣的一段时间里，在深度学习的一个视频中偶然看到Networkx这个库，它内置了常用的图与复杂网络分析算法，可以方便的进行复杂网络数据分析、仿真建模等工作。对于本次实验图的可视化这部分，Networkx+Matplotlib是一个很好的选择，操作方便而且功能多样。因此我选用了Networkx+Matplotlib作为图的可视化呈现。

## 提交内容说明

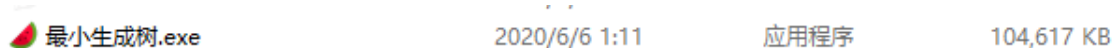
X01814031\_王浩宇\_最小生成树(基于kruscal算法+python实现).docx 为我的实验报告

最小生成树.exe 为我的执行文件，文件大小100M，这么大的原因是因为我使用了pyinstaller作为打包exe的工具，它的好处是可以把python环境和库放进去一起打包，因此不会受电脑环境的问题而无法执行，缺点就是存储占用有些大。

Kruscal\_Visualization.py 为我的Python代码。

## 运行方法:

### 1.首先打开 最小生成树.exe



然后打开后会出现这个画面



## 2.等待一会出现如下界面

Minimum Spanning Tree(kruscal)

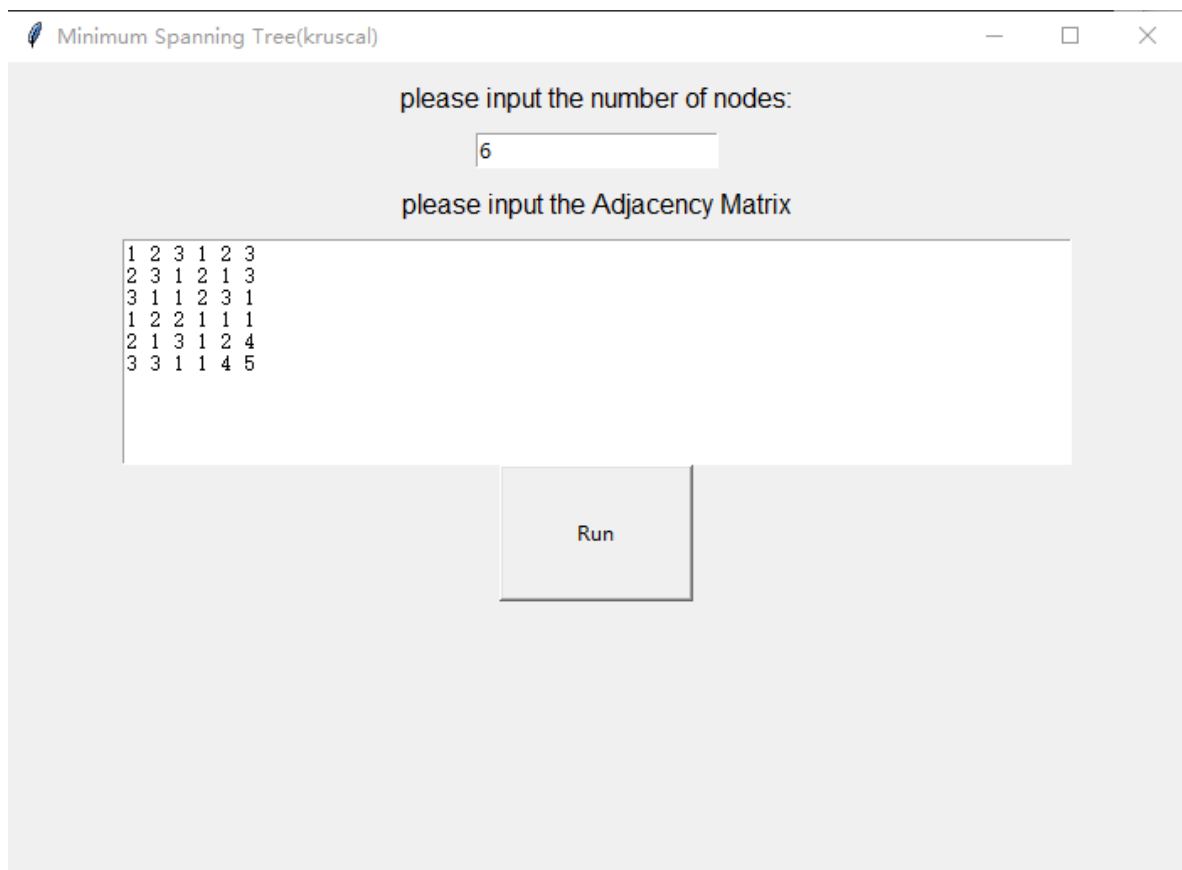
please input the number of nodes:

please input the Adjacency Matrix

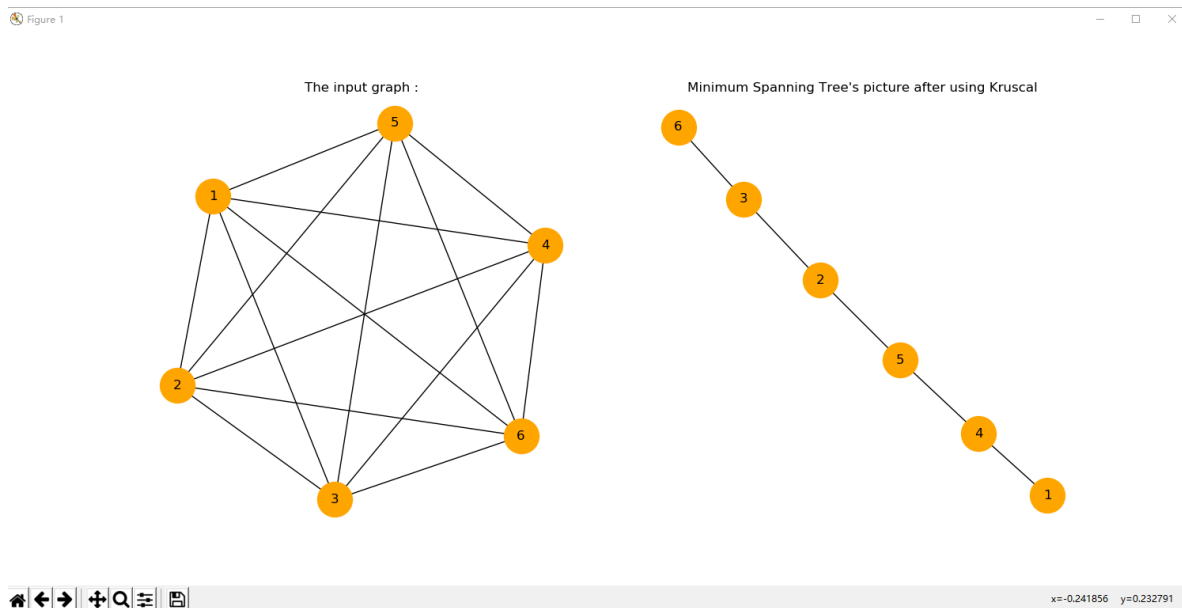
Run

这个就是我的应用界面了，然后有两个输入框，第一个输入框Please input the number of nodes是输入一个整数，代表有多少个点。第二个框Please input the Adjacency Matrix是输入邻接矩阵，该矩阵大小为 $n*n$ ，请在每行用空格隔开输入的数字(0代表没有路径)。

例如：



### 3.然后点击Run即可运行



其中左边的图代表输入邻接矩阵所形成的图，右边是生成的最小生成树的图

**注意：**由于我的水平有限，Tkinter的command命令循环调了很久也没有调好，程序每次打开只能运行一次，如果需要再次运行，请关闭重新打开。

谢谢阅读