**基于web的离散数学可视化演示系统**

1. **系统说明**

本项目开发前后端分离，前端采用 Vue 框架，后端采用 Spring Boot 框架。

1. **配置环境**
2. 前端：
3. 软件需求：VS Code、Node.js
4. 工具需求：vite、vue-router、element plus、echarts、axios
5. 配置过程：

①安装完需要的软件后，打开 VS Code 终端。

②输入指令：npm create vue@latest，安装 vite 脚手架，项目名称填 dm，除了 vue router 需要安装，其余选否。

③输入指令：npm i element-plus，安装 Element Plus。

④输入指令：npm i echarts -- save，安装 ECharts。

⑤输入指令：npm i axios，安装 axios。

1. 后端：

安装JDK。

1. **运行项目**
2. 前、后端分别运行：

后端直接运行即可，前端则需要打开项目终端，输入指令：npm run dev，点击链接打开 web 应用。

1. 通过 jar 包运行：

打开命令提示符，进入 jar 包所在的路径，输入 java -jar lisan-0.0.1-SNAPSHOT.jar，项目启动后，浏览器访问 http://localhost:8080 即可打开 web 应用。

1. **项目结构**
2. 前端主要文件夹和文件：

/src

/assets // 存放静态资源，如图片、样式文件

/components // 存放子组件，包括 dijkstra 算法和最小生成树的输入表单组件

/pages // 存放页面视图，每一个组件对应一个功能

/router // 存放 Vue 路由配置

/utils // 存放各种工具配置，包括用于前后端交互的 axios

App.vue // 根组件

main.js // 入口文件

1. 后端：

/src

/main

/java

/comlisanshuxue

/lisan

/chuandibibao //存放warshall算法功能

/djstl //存放dijsktra功能

/huff //存放huffman功能

/jihe //存放集合功能

/MingTi //存放真值表和SAT求解功能

/shengchengshu //存放最小生成树功能