平面最近点对问题 - 分治算法可视化

项目说明

本项目实现了平面最近点对问题的分治算法，并生成完整的可视化动画，满足算法设计与分析课程作业的所有要求。

文件结构

```
.
├── closest_pair_visualization.py    # 主程序（算法实现+可视化）
├── closest_pair_animation.py         #可视化动画
├── data_generator.py                 # 数据生成脚本
├── requirements.txt                  # 依赖包列表
├── README.pdf                         # 本文件
└── report_template.pdf                # 报告
```

环境要求

- Python 3.8.20 或更高版本
- 操作系统：Windows / macOS / Linux

依赖安装

核心依赖

```
pip install -r requirements.txt
```

或：

```
pip install numpy matplotlib pillow
```

这三个包足以完成所有功能，无需其他外部工具！

注意：默认配置使用 GIF 格式，完全离线运行，无需 ffmpeg！

运行方法

1. 一键运行（完全离线）

```
# 安装依赖（仅需一次）
pip install numpy matplotlib pillow

# 运行程序（自动生成 GIF 动画）
python closest_pair_visualization.py
```

程序将：

- 生成满足要求的随机点集
- 执行分治算法求解最近点对
- 自动生成可视化动画文件 `closest_pair_beautiful_final.gif`

2. 自定义参数

在 `closest_pair_visualization.py` 的 `main()` 函数中修改参数：

```
N = 45          # 点的数量（要求 >= 36）
SEED = 42       # 随机种子（保证可复现）
OUTPUT_FILE = 'closest_pair_animation.gif'  # GIF 格式
FPS = 15        # 动画帧率
```

如果需要 MP4 格式（需要安装 ffmpeg）：

```
OUTPUT_FILE = 'closest_pair_animation.mp4'
```

3. 生成测试数据

```
python data_generator.py
```

可生成多组测试数据，用于验证算法正确性。
注 主程序中所用的随机数据在 `closest_pair_visualization.py` 中直接生成，不是在这个文件中生成。

算法特点

分治策略

1. **预处理**：按 x 坐标排序点集
2. **分割**：用垂直线将点集分为左右两部分
3. **递归求解**：分别求左右两侧的最近点对
4. **合并**：检查跨越分割线的点对，更新全局最小距离

时间复杂度

- **预处理排序**：O(n log n)
- **递归求解**：T(n) = 2T(n/2) + O(n)
- **总时间复杂度**：O(n log n)

空间复杂度

- O(n)（主要用于存储排序后的点集和递归栈）

可视化特性

动画展示以下关键步骤：

1. **初始点集**：显示随机生成的点分布
2. **排序预处理**：按 x 坐标排序
3. **递归分割**：显示分割线和左右子问题
4. **基本情况**：小规模问题的暴力求解
5. **带区检查**：高亮显示带区及带区内的点
6. **最优点对更新**：动态显示当前找到的最近点对
7. **最终结果**：标注最终的最近点对和距离

数据生成策略

满足作业要求：

- ✅ 点数 n ≥ 36
- ✅ 点位于分界线两侧，左右点数相差 10%-20%
- ✅ 在中线±10%宽度内至少有 30% 的点

- ✅ 提供随机种子保证可复现

输出文件

- **closest_pair_beautiful_final.gif**：可视化动画
- 动画时长：约 60 秒
- 分辨率：1200x1000（>720p）

如果需要更高质量，可选择 MP4 格式（需安装 ffmpeg）。

验证结果

程序运行后会在控制台输出：

- 最小距离
- 最近点对的坐标
- 动画生成状态

常见问题

Q: 是否需要安装 ffmpeg?

A: **不需要！** 默认使用 GIF 格式，只需要 Python 的三个包（numpy, matplotlib, pillow）即可完全离线运行。

Q: 动画生成失败?

A:

1. 确认已安装所有依赖
2. 检查磁盘空间是否足够
3. 尝试降低 dpi 或 fps

Q: 如何加快/减慢动画速度?

A: 修改 `FPS` 参数，值越大动画越快