初稿

function[]=Mapof4color(prov,se)

map=shaperead('省界\_region.shp');

N=length(map);

A=zeros(N);%A是邻接矩阵，存储各省的边界接壤信息

for i=1:N

shapei=[map(i).X(:),map(i).Y(:)];

for j=(i+1):N;

shapej=[map(j).X(:),map(j).Y(:)];

if size(intersect(shapei,shapej,'rows'),1)>=2

A(i,j)=1;

A(j,i)=1;

end

end

end

color=Pic(prov,se,A,N);

colour=[1,0,0;0,1,0;0,0,1;1,1,0];

figure;

hold on

for i=1:N

index=find(isnan(map(i).X));

start=1;

for j=1:length(index)

fill(map(i).X(start:(index(j)-1)),map(i).Y(start:(index(j)-1)),colour(color(i),:));

start=index(j)+1;

end

end

function [color] = Pic(prov,se,A,N)

%UNTITLED6 此处显示关此函数的摘要

% 此处显示详细说明

color=zeros(1,N);%color存储每一个省的颜色信息，颜色有1，2，3，4四种序号，初始时全赋零值

color(prov)=se;

fag=0;%fag判断是不是异常而返回上一次变量k的循环。若fag=1，说明col=5而需要调整前一次变量k的循环选定的col

for k=1:N

if(k==prov&&fag==0)

continue;

elseif(k==prov&&fag==1)

k=k-2;continue;

end

col=1;

tag=0;%判断testcolor函数输出的值，若col=5，则tag=1，该省无色可选

if(fag==0)

col=testcolor(col,k,A);%运用testcolor函数来选择出不与接壤省颜色冲突的颜色

else

col=testcolor(col+1,k,A);%fag=1，说明此时程序是从上一个“无色可选”的k的循环中回来的，为保证之后的k的循环有色可选，需要改变这一个省的颜色，故col加1，测试col+1行不行

end

if(col==5)

tag=1;%哪种颜色都不行

end

if(tag==1)

k=k-2;fag=1;continue; %这一个省出现了无色可选的情况，需要回到之前一次的k的循环，令fag=1

else

color(k)=col;fag=0;%该省有色可选

end

end

end

function [ col ] = testcolor( col,k,A )

%UNTITLED4 此处显示有关此函数的摘要

% 此处显示详细说明

for i=1:k

if(A(k,i)==1&&col==color(i))%看这个pro省此时待定的颜色是否与接壤省的颜色相冲突

col=col+1;i=0;%重新选下一个色，从i=1开始再判断此时的col是否与接壤省的颜色冲突

end

if(col>4) %哪种颜色都不行

break;

end

end

return;%循环正常结束，说明找到了一个不与任何接壤省冲突的颜色，并返回该颜色

end

二级

function [color] = Pic(prov,se,A,N)

%UNTITLED6 此处显示关此函数的摘要

% 此处显示详细说明

color=ones(1,N);%color存储每一个省的颜色信息，颜色有1，2，3，4四种序号，初始时全赋零值

color(prov)=se;

fag=0;%fag判断是不是异常而返回上一次变量k的循环。若fag=1，说明col=5而需要调整前一次变量k的循环选定的col

k=1;

while(k<=N)

if(k==prov&&fag==0)

k=k+1;

elseif(k==prov&&fag==1)

k=k-1;

end

col=1;

if(fag==0)

col=testcolor(col,k,A);%运用testcolor函数来选择出不与接壤省颜色冲突的颜色

else

col=testcolor(color(k)+1,k,A);%fag=1，说明此时程序是从上一个“无色可选”的k的循环中回来的，为保证之后的k的循环有色可选，需要改变这一个省的颜色，故col加1，测试col+1行不行

end

if(col==5)

k=k-1;

fag=1; %这一个省出现了无色可选的情况，需要回到之前一次的k的循环，令fag=1

else

color(k)=col;fag=0;k=k+1;%该省有色可选

end

end

end

function [ col ] = testcolor( col,k,A )

%UNTITLED4 此处显示有关此函数的摘要

% 此处显示详细说明

i=1;

while(i<=k)

if(A(k,i)==1&&col==color(i))%看这个pro省此时待定的颜色是否与接壤省的颜色相冲突

col=col+1;i=0;%重新选下一个色，从i=1开始再判断此时的col是否与接壤省的颜色冲突

end

if(col>4) %哪种颜色都不行

break;

end

i=i+1;

end

return;%循环正常结束，说明找到了一个不与任何接壤省冲突的颜色，并返回该颜色

end

惊人的终稿（弃用testcolor函数）

（怀疑之前用testcolor不行是因为其return的值不正确，待查明原因）

function [color] = Pic(prov,se,A,N)

%UNTITLED6 此处显示关此函数的摘要

% 此处显示详细说明

color=ones(1,N);%color存储每一个省的颜色信息，颜色有1，2，3，4四种序号，初始时全赋零值

color(prov)=se;

fag=0;%fag判断是不是异常而返回上一次变量k的循环。若fag=1，说明col=5而需要调整前一次变量k的循环选定的col

k=1;

while(k<=N)

if(k==prov&&fag==0)

k=k+1;

elseif(k==prov&&fag==1)

k=k-1;

end

if(fag==0)

i=1;col=1;

while(i<=k)

if(A(k,i)==1&&col==color(i))%看这个pro省此时待定的颜色是否与接壤省的颜色相冲突

col=col+1;i=0;%重新选下一个色，从i=1开始再判断此时的col是否与接壤省的颜色冲突

end

if(col>4) %哪种颜色都不行

break;

end

i=i+1;

end

%col=testcolor(1,k,A);%运用testcolor函数来选择出不与接壤省颜色冲突的颜色

else

i=1;col=color(k)+1;

while(i<=k)

if(A(k,i)==1&&col==color(i))%看这个pro省此时待定的颜色是否与接壤省的颜色相冲突

col=col+1;i=0;%重新选下一个色，从i=1开始再判断此时的col是否与接壤省的颜色冲突

end

if(col>4) %哪种颜色都不行

break;

end

i=i+1;

end

%col=testcolor(color(k)+1,k,A);%fag=1，说明此时程序是从上一个“无色可选”的k的循环中回来的，为保证之后的k的循环有色可选，需要改变这一个省的颜色，故col加1，测试col+1行不行

end

if(col==5)

k=k-1;

fag=1; %这一个省出现了无色可选的情况，需要回到之前一次的k的循环，令fag=1

else

color(k)=col;fag=0;k=k+1;%该省有色可选

end

end

end

颜色序号：1. 红 2. 绿 3. 蓝 4. 黄 5.天蓝色

省：

1 浙江 1 4 错误1

2 云南

3 新疆

4 香港

5 西藏

6 台湾 1 错误2

7 四川

8 陕西 3 4 错误1

9 山西 3 4 错误3

10 山东 1 4 错误3

11 青海 2 错误3

12 宁夏 3 错误3

13 内蒙古 4 错误3

14 辽宁 1 3 错误3

15 江西 1 错误3

16 吉林 3 错误3

17 湖南 2 3 4 错误3

18 湖北 4 错误3

19 黑龙江 3 错误3

20 河南 2 错误3

21 北京 1 错误3

22 天津 1 错误3

23 海南 4 错误2

24 贵州 3 4 错误3

25 广西

26 甘肃 1 2 3 错误3

27 福建 1 错误2

28 澳门

29 安徽 2 3 4 错误3

30 上海 3 错误3

31 重庆

32 江苏 3 错误3

33 广东 1 错误3

34 河北

大概估计总共37处错误，犯错率27%

其中错误3有30处，无法四色的概率为22%

错误1：下标索引必须为正整数类型或逻辑类型。出错 Mapof4color (line 58) i=1;col=color(k)+1; %fag=1，说明之后的省无色可填，需要调整本省的颜色，从第color(k)+1个开始检测

错误2：出现相邻省重色

错误3：使用了第五种颜色

人机交互

function[]=Mapof4color(se)

mapof4color=shaperead('省界\_region.shp');

N=length(mapof4color); %中国地图所包含的省（或其他同级行政区）的数量

A=zeros(N); %A是邻接矩阵，存储各省的边界接壤信息

for i=1:N %读取地图信息，填充邻接矩阵

shapei=[mapof4color(i).X(:),mapof4color(i).Y(:)];

for j=(i+1):N;

shapej=[mapof4color(j).X(:),mapof4color(j).Y(:)];

if size(intersect(shapei,shapej,'rows'),1)>=2 %通过intersect函数判断两省之间是否存在的共同点大于2

A(i,j)=1; %若是，则邻接矩阵相应元素为1

A(j,i)=1;

end

end

end

geoshow('省界\_region.shp');

[x0,y0] = ginput(1);

for i=1:N %读取地图信息，填充邻接矩阵

% shapei=[mapof4color(i).X(:),mapof4color(i).Y(:)];

for j=(i+1):N;

% shapej=[mapof4color(j).X(:),mapof4color(j).Y(:)];

[in,~]= inpolygon(x0,y0,mapof4color(i).X(:),mapof4color(i).Y(:));

if(in==1)

prov=i;

end

end

end

color=Pic(prov,se,A,N);

colour=[1,0,0;0,1,0;0,0,1;1,1,0];

figure;

hold on

for i=1:N

index=find(isnan(mapof4color(i).X));

start=1;

for j=1:length(index)

fill(mapof4color(i).X(start:(index(j)-1)),mapof4color(i).Y(start:(index(j)-1)),colour(color(i),:));

start=index(j)+1;

end

end

1 黑龙江

2 吉林

3 辽宁

4 内蒙古

5 河北

6 北京

7 天津

8 山西

9 陕西

10 宁夏

11 甘肃

12 新疆

13 青海

14 西藏 1 错误2、错误3

15 四川

16 重庆

17 湖北

18 河南

19 山东

20 江苏

21 安徽

22 上海

23 浙江

24 江西

25 湖南

26 贵州

27 云南

28 广西 2 错误2、错误3

29 广东

30 香港

31 澳门（用点击型交互式无法完成，因为图片太小无法放大；考虑增加文字型输入方式）

32 福建

33 台湾

34 海南

犯错率约为1.5%，五色概率为1.5%，且五色情况也不符合“相邻省色不同”的要求，错误情况十分严重。