* MAIN\_Final.m

clear;

clc;

%--------------------------------------------------------------------------

%程序计算模型定义，包括构件几何参数，单元参数，材料参数，均布荷载大小。

%程序单位采用国际单位制基本单位，长度单位：m，力的单位：N，应力单位：Pa。

L = 10.0;%m 构件长度

H = 2.0;%m 构件高度

t=0.01; %m %单元厚度

ele\_length = 0.5;%m 单元长度

ele\_height = 0.5;%m 单元高度

E=2.1e11; %Pa 材料弹性模量

miu=0.3; %泊松比

load\_q=20000; %N/m 均布荷载

node\_load=load\_q\*ele\_length; %等效结点荷载

%--------------------------------------------------------------------------

%--------------------------------------------------------------------------

%网格划分

ele\_num\_L = L/ele\_length; %长度方向单元个数

ele\_num\_H = H/ele\_height; %宽度方向单元个数

node\_num\_L = ele\_num\_L + 1; %长度方向节点个数

node\_num\_H = ele\_num\_H + 1; %宽度方向节点个数

node = []; %定义节点列表

ele = []; %定义单元列表

%生成节点列表 节点列表结构为 [节点编号 X坐标 Y坐标 Z坐标（平面问题，该项为0）]

for j = 1:node\_num\_H

for i = 1 : node\_num\_L

node\_new = [i+(j-1)\*node\_num\_L ele\_length\*(i-1) ele\_height\*(j-1) 0];

node = cat (1,node,node\_new);

end

end

%生成单元列表 单元列表结构为[单元编号 第一结点编号 第二节点编号 第三节点编号]（左下角为起点，逆时针顺序）

for j = 1:ele\_num\_H

for i = 1 : ele\_num\_L

ele\_new\_odd = [2\*i-1+2\*(j-1)\*ele\_num\_L i+(j-1)\*(ele\_num\_L+1) i+1+(j-1)\*(ele\_num\_L+1) i+ node\_num\_L+1+(j-1)\*(ele\_num\_L+1)];

ele\_new\_even = [2\*i+2\*(j-1)\*ele\_num\_L i+(j-1)\*(ele\_num\_L+1) i+(j-1)\*(ele\_num\_L+1)+ node\_num\_L+1 i+(j-1)\*(ele\_num\_L+1)+ node\_num\_L];

ele = cat (1,ele,ele\_new\_odd);

ele = cat (1,ele,ele\_new\_even);

end

end

%--------------------------------------------------------------------------

%生成单元刚度矩阵

%基础参数与单元定义

num\_ele=size(ele,1);

n\_ele=length(ele(:,1)); %单元数

dof=length(node(:,1))\*2; %自由度，梁单元的每个节点有2个自由度，横向位移，纵向位移

f=zeros(dof,1); %整体坐标系下结构整体外荷载矩阵

f\_loc=zeros(6,1); %单元外荷载矩阵，局部坐标系下

u=ones(dof,1); %矩阵位移

K=zeros(dof); %总体刚度矩阵

stress=zeros(n\_ele,1); %单元应力矩阵

%生成总体刚度矩阵

for i=1:n\_ele

k\_ele=TriangleElementStiffness(E,miu,t,node(ele(i,2:4),2:4));

K=assemTriangle(K,k\_ele,ele(i,2),ele(i,3),ele(i,4));

end

%--------------------------------------------------------------------------

%定义边界条件

%力边界条件

for i=node\_num\_L\*(node\_num\_H-1): node\_num\_L\*node\_num\_H

f(2\*i)=-node\_load;%N 顶部

end

f(2\*node\_num\_L\*(node\_num\_H-1)) = -node\_load/2;%N 左右边缘两点

f(2\*node\_num\_L\*(node\_num\_H)) = -node\_load/2;%N 左右边缘两点

% 位移边界条件；

u(1)=0;

u(2)=0;

%u(2\*node\_num\_L-1)=0;

u(2\*node\_num\_L)=0;

%--------------------------------------------------------------------------

%求解

%求解未知自由度

index=[]; %未知自由度索

p=[]; %位置自由度对应的节点力矩阵matrix；

for i =1:dof

if u(i)~=0

index=[index,i];

p=[p;f(i)];

end

end

%求解位移矩阵

u(index)=K(index,index)\p;

%计算节点变形后位置（为画图明显，对位移进行20倍放大

x1=node(:,2)+20\*u(1:2:node\_num\_L\*node\_num\_H\*2);

y1=node(:,3)+20\*u(2:2:node\_num\_L\*node\_num\_H\*2);

%--------------------------------------------------------------------------

%应力求解及画图

stress=zeros(num\_ele,3);

%mises应力图

figure;

mises\_stress=zeros(num\_ele);

for i=1:n\_ele

u1=[u(2\*ele(i,2)-1);u(2\*ele(i,2));u(2\*ele(i,3)-1);u(2\*ele(i,3));u(2\*ele(i,4)-1);u(2\*ele(i,4))];

stress(i,:)=TriangleElementStress(E,miu,node(ele(i,2:4),2:3),u1,1)'; %单元应力

mises\_stress(i)=((stress(i,1)+stress(i,2))^2-3\*(stress(i,1)\*stress(i,2)-stress(i,3)^2))^0.5;%mises应力

patch(node(ele(i,2:4),2),node(ele(i,2:4),3),mises\_stress(i));

end

set(get(colorbar,'title'),'string','\fontname{Times New Roman}\fontsize{14}Pa')

title('\fontname{宋体}\fontsize{14}单元应力\fontname{Times New Roman}\fontsize{14}(Mises)')

%X方向主应力图

figure;

for i=1:n\_ele

patch(node(ele(i,2:4),2),node(ele(i,2:4),3),stress(i,1));

end

colormap jet;

colorbar;

set(get(colorbar,'title'),'string','Pa');

title('\fontname{宋体}\fontsize{14}单元应力\fontname{Times New Roman}\fontsize{18}(\sigma\_x)');

%Y方向主应力图

figure;

for i=1:n\_ele

patch(node(ele(i,2:4),2),node(ele(i,2:4),3),stress(i,2));

end

colormap jet;

colorbar;

set(get(colorbar,'title'),'string','Pa');

title('\fontname{宋体}\fontsize{14}单元应力\fontname{Times New Roman}\fontsize{18}(\sigma\_y)');

%剪应力图

figure;

for i=1:n\_ele

patch(node(ele(i,2:4),2),node(ele(i,2:4),3),stress(i,3));

end

hold on;

colormap jet;

colorbar;

set(get(colorbar,'title'),'string','Pa');

title('\fontname{宋体}\fontsize{14}单元应力\fontname{Times New Roman}\fontsize{18}(\tau\_x)');

hold on;

%变形图

figure;

for i=1:n\_ele

patch(node(ele(i,2:4),2),node(ele(i,2:4),3),'w','FaceColor','none','LineStyle','-')

hold on;

patch(x1(ele(i,2:4)),y1(ele(i,2:4)),'w','FaceColor','none','EdgeColor','r');

end

legend('\fontname{宋体}\fontsize{14}初始形态','\fontname{宋体}\fontsize{14}加载形态')

* TriangleElementStiffness.m

function k\_ele=TriangleElementStiffness(E,miu,t,node\_ele)

% 三角单元刚度矩阵

x1=node\_ele(1,1);

y1=node\_ele(1,2);

x2=node\_ele(2,1);

y2=node\_ele(2,2);

x3=node\_ele(3,1);

y3=node\_ele(3,2);

% ---------------------------

A=(x1\*(y2-y3)+x2\*(y3-y1)+x3\*(y1-y2))/2; %单元面积

a1=x2\*y3-y2\*x3;

a2=y1\*x3-x1\*y3;

a3=x1\*y2-y1\*x2;

b1=y2-y3;

b2=y3-y1;

b3=y1-y2;

c1=x3-x2;

c2=x1-x3;

c3=x2-x1;

D=E/(1-miu^2)\*[1 miu 0;

miu 1 0;

0 0 (1-miu)/2];

B1 = [b1 0;0 c1;c1 b1];

B2 = [b2 0;0 c2;c2 b2];

B3 = [b3 0;0 c3;c3 b3];

B\_single\_element = [B1 B2 B3]./2./A;

k\_ele = B\_single\_element'\*D\*B\_single\_element.\*t.\*A;

* TriangleElementStress.m

function str=TriangleElementStress(E,miu,node\_ele,u1,p)

% 三角单元应力矩阵

x1=node\_ele(1,1);

y1=node\_ele(1,2);

x2=node\_ele(2,1);

y2=node\_ele(2,2);

x3=node\_ele(3,1);

y3=node\_ele(3,2);

% ---------------------------

A=(x1\*(y2-y3)+x2\*(y3-y1)+x3\*(y1-y2))/2; %为单元的面积

a1=x2\*y3-y2\*x3;

a2=y1\*x3-x1\*y3;

a3=x1\*y2-y1\*x2;

b1=y2-y3;

b2=y3-y1;

b3=y1-y2;

c1=x3-x2;

c2=x1-x3;

c3=x2-x1;

B=1/2/A\*[b1 0 b2 0 b3 0;

0 c1 0 c2 0 c3;

c1 b1 c2 b2 c3 b3];

if p==1

D=E/(1-miu^2)\*[1 miu 0;

miu 1 0;

0 0 (1-miu)/2];

elseif p==2

D=E/(1+miu)/(1-2\*miu)\*[1-miu miu 0;

miu 1-miu 0;

0 0 (1-2\*miu)/2];

end

str=D\*B\*u1; %为单元应力的计算

* assemTriangle.m

function k\_t=assemTriangle(k\_t,k\_ele,node1,node2,node3)

% 组装总刚

d(1:2)=2\*node1-1:2\*node1;

d(3:4)=2\*node2-1:2\*node2;

d(5:6)=2\*node3-1:2\*node3;

for ii=1:6

for jj=1:6

k\_t(d(ii),d(jj))=k\_t(d(ii),d(jj))+k\_ele(ii,jj);

end

end