```
1  //ノードの描画
2  glPointSize(10);
3  glBegin(GL_POINTS);
4  for (i = 0; i < _mesh->num_node; i++) {
5      //頂点の状態に応じて色を変える
6      switch (_mesh->node[i].state) {
7      case NODE_FREE:
8          glColor3d(0, 1, 1);
9          break;
10     case NODE_FIXED:
11         glColor3d(1, 0, 1);
12         break;
13     case NODE_DEFORM:
14         glColor3d(1, 1, 0);
15         break;
16     }
17     //頂点の座標を指定
18     glVertex3dv(_mesh->node[i].new_position.X);
19 }
20 glEnd();
21
22 //ノード間ラインの描画(変形前・変形後)
23 for (i = 0; i < _mesh->num_tetrahedra; i++) {
24     for (j = 0; j < 4; j++) {
25         for (k = 0; k < j; k++) {
26             glLineWidth(0.5);
27             glColor3d(0.3, 0.3, 0.3);
28             glBegin(GL_LINE_STRIP);
29             glVertex3dv(_mesh->tetrahedra[i].position[j].X);
30             glVertex3dv(_mesh->tetrahedra[i].position[k].X);
31             glEnd();
32
33             glLineWidth(1);
34             glColor3d(0, 0, 0);
35             glBegin(GL_LINE_STRIP);
36             glVertex3dv(_mesh->tetrahedra[i].new_position[j].X);
37             glVertex3dv(_mesh->tetrahedra[i].new_position[k].X);
38             glEnd();
39         }
40     }
41 }
42
43 //要素ポリゴンの描画
44 glColor3d(0, 0, 1);
45 for (i = 0; i < _mesh->num_tetrahedra; i++) {
46     calColorMap(_mesh->tetrahedra[i].mises_stress / _max_mises_stress, &color);
47     for (j = 0; j < 4; j++) {
48         glColor3dv(color.X);
49         glBegin(GL_TRIANGLES);
50         glVertex3dv(_mesh->tetrahedra[i].new_position[(j + 0) % 4].X);
51         glVertex3dv(_mesh->tetrahedra[i].new_position[(j + 1) % 4].X);
52         glVertex3dv(_mesh->tetrahedra[i].new_position[(j + 2) % 4].X);
53         glEnd();
54     }
55 }
```