

gki3Object.java

```
package gobalkrishnan_v_18_06_1995.graphics;

import gobalkrishnan_v_18_06_1995.frame.gkiFrame;

public class gki3Object {

    ArrayList<gki3Polygon> polygon=new ArrayList<>();
    public ArrayList<gkiPixel> pixel=new ArrayList<>();
    gkiFrame frame;
    gkiPerspective perspective=new gkiPerspective();
    public gki3Object(gkiFrame f){
        frame=f;
    }

    public void gki3Polygon(gki3Polygon p){
        polygon.add(p);
    }
    public void gki3Polygon(gki3Polygon[] p){
        for(int i=0;i<p.length;i++){
            polygon.add(p[i]);
        }
    }
    public void gki3Polygon(ArrayList<gki3Polygon> p){
        for(int i=0;i<p.size();i++){
            polygon.add(p.get(i));
        }
    }

    public void tranlate(gki3Point p){
```

gki3Object.java

```
for(int i=0;i<polygon.size();i++){
    for(int j=0;j<polygon.get(i).points.size();j++){
        polygon.get(i).points.get(j).add(p);
    }
}

}

public void scale(gki3Point p){
    for(int i=0;i<polygon.size();i++){
        for(int j=0;j<polygon.get(i).points.size();j++){
            polygon.get(i).points.get(j).scale(p);
        }
    }
}

public void rotateX(gki3Point c,double a){
    for(int i=0;i<polygon.size();i++){
        for(int j=0;j<polygon.get(i).points.size();j++){
            polygon.get(i).points.get(j).rotateX(c, a);;
        }
    }
}

public void rotateY(gki3Point c,double a){
    for(int i=0;i<polygon.size();i++){
        for(int j=0;j<polygon.get(i).point.size();j++){
            polygon.get(i).point.get(j).rotateY(c, a);;
        }
    }
}

public void rotateZ(gki3Point c,double a){
    for(int i=0;i<polygon.size();i++){
        for(int j=0;j<polygon.get(i).point.size();j++){
```

gki3Object.java

```
        polygon.get(i).point.get(j).rotateZ(c, a);  
    }  
}  
}
```

```
ArrayList<gki3Point> poin=new ArrayList<>();  
ArrayList<ArrayList<gki3Point>> poly=new ArrayList<>();  
ArrayList<gki3Point> gs=new ArrayList<>();  
ArrayList<gki3Line> gline=new ArrayList<>();
```

```
public void zdepth(){  
    gline.removeAll(gline);  
    for(int i=0;i<polygon.size();i++){  
        polygon.get(i).process();  
        for(int j=0;j<polygon.get(i).point.size();j++){  
            gki3Point pt=polygon.get(i).point.get(j);  
            poin.add(pt);  
        }  
    }  
}
```

```
zsort(poin);  
int c=0;
```

```
    if(polygon.size()>1){
```

```
        boolean f=false;  
        ArrayList<gki3Point> pi=new ArrayList<>();  
        for(int i=0,j=1;i<poin.size();i++,j++){
```

gki3Object.java

```
if(j==poin.size()){j=0;}

gki3Point u=poin.get(i);
gki3Point v=poin.get(j);

// System.out.println(u+":"+v);
if(u.z==v.z){
    pi.add(u);
    //pi.add(e)
}else{

    if(c==0){
        pi.add(u);
        poly.add(pi);
    }
    pi=new ArrayList<>();
    poly.add(pi);
    pi.add(u);

    // System.out.println(u+":");
    // System.out.println(c);
    c++;
    //System.out.println("  ");
}

}
// System.out.println(poin.size());
for(int i=0,j=1;i<poly.size()-1;i++,j++){
    if(j==poly.size()){
        j=0;
```

gki3Object.java

```
    }
    int k=i+1;
    if(i>0){
        poly.get(i).remove(0);
        poly.get(i).add(poly.get(k).get(0));
    }

}
}else{
    poly.add(poin);
}

//
//gki3Line ls=new gki3Line();
//for(int i=0;i<poly.size();i++){
// if(poly.get(i).size()>1){
//     int k=0;
//     int r=0;
//     StringBuilder s=new StringBuilder();
//
//     for(int j=0;j<poly.get(i).size();j++){
//         if(k%2==0){
//             ls=new gki3Line();
//             ls.sp(poly.get(i).get(k));
//             s.append(k+":");
//         }else{
//             ls.ep(poly.get(i).get(k));
//             ls.process();
//             gline.add(ls);
//             s.append(k+"\n");
//             r++;
//         }
//     }
//     //System.out.println(ls);
```

gki3Object.java

```
//  
// }  
// k++;  
// }  
// //System.out.println(r);  
//     System.out.println(" ");  
//  
//  
// //System.out.println(poly.get(i).size());  
//}  
//}
```

```
//System.out.println(gline.size()+":");
```

```
}
```

```
private void zsort(ArrayList<gki3Point> a){  
    gki3Point[] l=new gki3Point[a.size()];  
    for(int i=0;i<l.length;i++){  
        l[i]=a.get(i);  
    }  
    gkiMergeSort ysort=new gkiMergeSort();  
    ysort.sortZdepthAscending(l);
```

```
    tolist(a, l);  
}
```

gki3Object.java

```
private void tolist(ArrayList<gki3Point> b,gki3Point[] a){
    b.removeAll(b);
    for(int i=0;i<a.length;i++){
        b.add(a[i]);
    }
}

public gki3Point mid(){
    double x = 0,y=0,z=0;
    double len=0;
    for(int i=0;i<polygon.size();i++){
        for(int j=0;j<polygon.get(i).points.size();j++){
            x+=polygon.get(i).points.get(j).x;
            y+=polygon.get(i).points.get(j).y;
            z+=polygon.get(i).points.get(j).z;

            len++;
        }
    }

    return new gki3Point(x/((double)len, y/((double)len, z/((double) len);
}

public ArrayList<gki2Polygon> gkiPerspective(){
    pixel.removeAll(pixel);
    zdepth();
    ArrayList<gki2Polygon> gg=new ArrayList<>();
    ArrayList<ArrayList<gki2Point>> gp=new ArrayList<>();
    gkiPerspective per=new gkiPerspective();
```

gki3Object.java

```
per.perspective(30,1, -50, 1000, -100, 100, -100, 100);
for(int j=0;j<poly.size();j++){
    ArrayList<gki2Point> y=new ArrayList<>();

    for(int i=0;i<poly.get(j).size();i++){
        gki2Point p= per.screen2d(poly.get(j).get(i));
        p.color(poly.get(j).get(i).color);
        p.scale(1000);
        p.translate(frame.getWidth()/2, frame.getHeight()/2);
        y.add(p);
    }
    gki2QuickHull t=new gki2QuickHull();
    ArrayList<gki2Point> w=t.quickhull(y);

    gp.add(w);
}

for(int i=0;i<gp.size();i++){
    //System.out.println(gp.get(i));

    gki2Polygon t=new gki2Polygon();
    for(int j=0;j<gp.get(i).size();j++){
        t.gki2Point(gp.get(i).get(j));
        t.gkiColor(gp.get(i).get(j).color);
    }

    t.processX();
    gg.add(t);
}
```


gki3Object.java

```
    return gg;
}

public ArrayList<gkiPixel> pix(){
    pixel.removeAll(pixel);
    zdepth();
    ArrayList<gkiPixel> gp=new ArrayList<>();
    gkiPerspective per=new gkiPerspective();
    per.perspective(30,1, -50, 1000, -100, 100, -100, 100);
    for(int i=0;i<poly.size();i++){
        gkiPixel t=new gkiPixel();
        for(int j=0;j<poly.get(i).size();j++){
            gki3Point e=poly.get(i).get(j);
            gki2Point p= per.screen2d(e);
            p.color(poly.get(i).get(j).color);
            p.scale(1000);
            p.translate(frame.getWidth()/2, frame.getHeight()/2);
            // if(p.color!=null)
            if(p.x!=Double.NaN && p.y!=Double.NaN){
                System.out.println(p);

                t.setARGB((int)p.x, (int)p.y,new gkiColor(0xffff0000));
            }
            gp.add(t);
        }
    }
    return gp;
}
```

gki3Object.java

}

public void gkiOrthogonal(){

}

}