

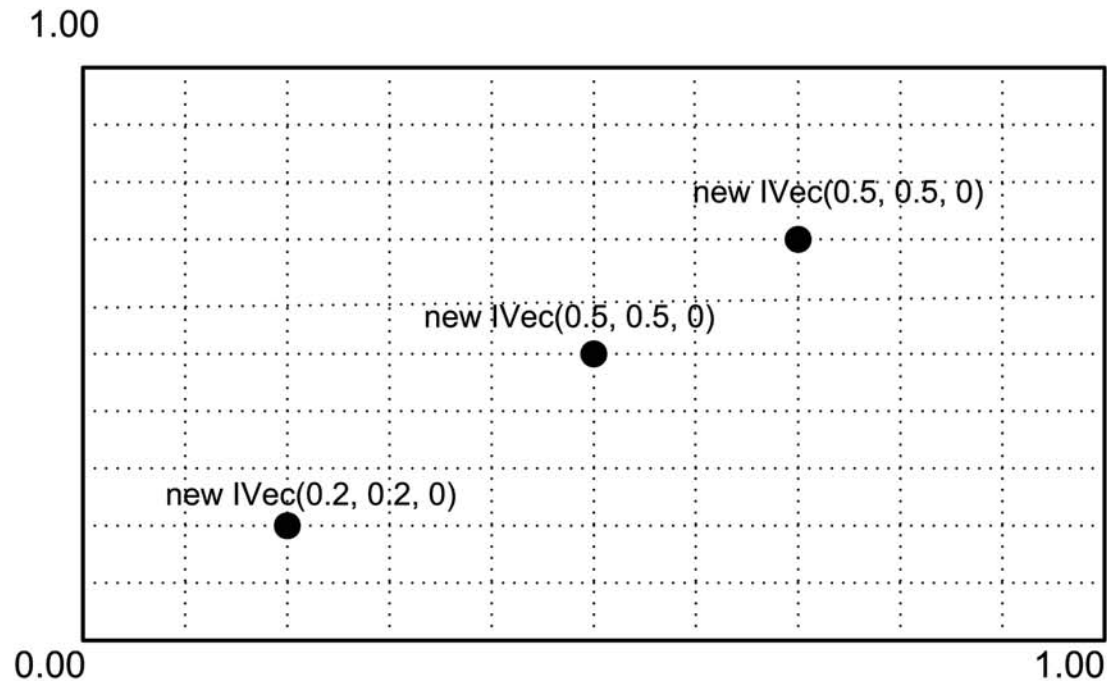


Advance Coding Form
Final

Amir Habibabadi

Instructor: Satoru Sugihara

1) Introducing the surface and starting point to the processing



```
IG.open("srf.3dm");
ISurface surf = IG.surface(0);
surf.del();
IG.fill();
new LineAgent(new IVec(0.2, 0.2, 0), new IVec(1, 0, 0), surf).clr(0.2);
new LineAgent(new IVec(0.5, 0.5, 0), new IVec(1, 0, 0), surf).clr(0.2);
new LineAgent(new IVec(0.5, 0.5, 0), new IVec(1, 0, 0), surf).clr(0.2);
}
```

2) Introducing the static class

```
static class LineAgent extends IAgent {
    static double length = 0.01;
    static double clearance = 0.0099;

    IVec pt1, pt2;
    boolean isColliding=false;
    ISurface surf;

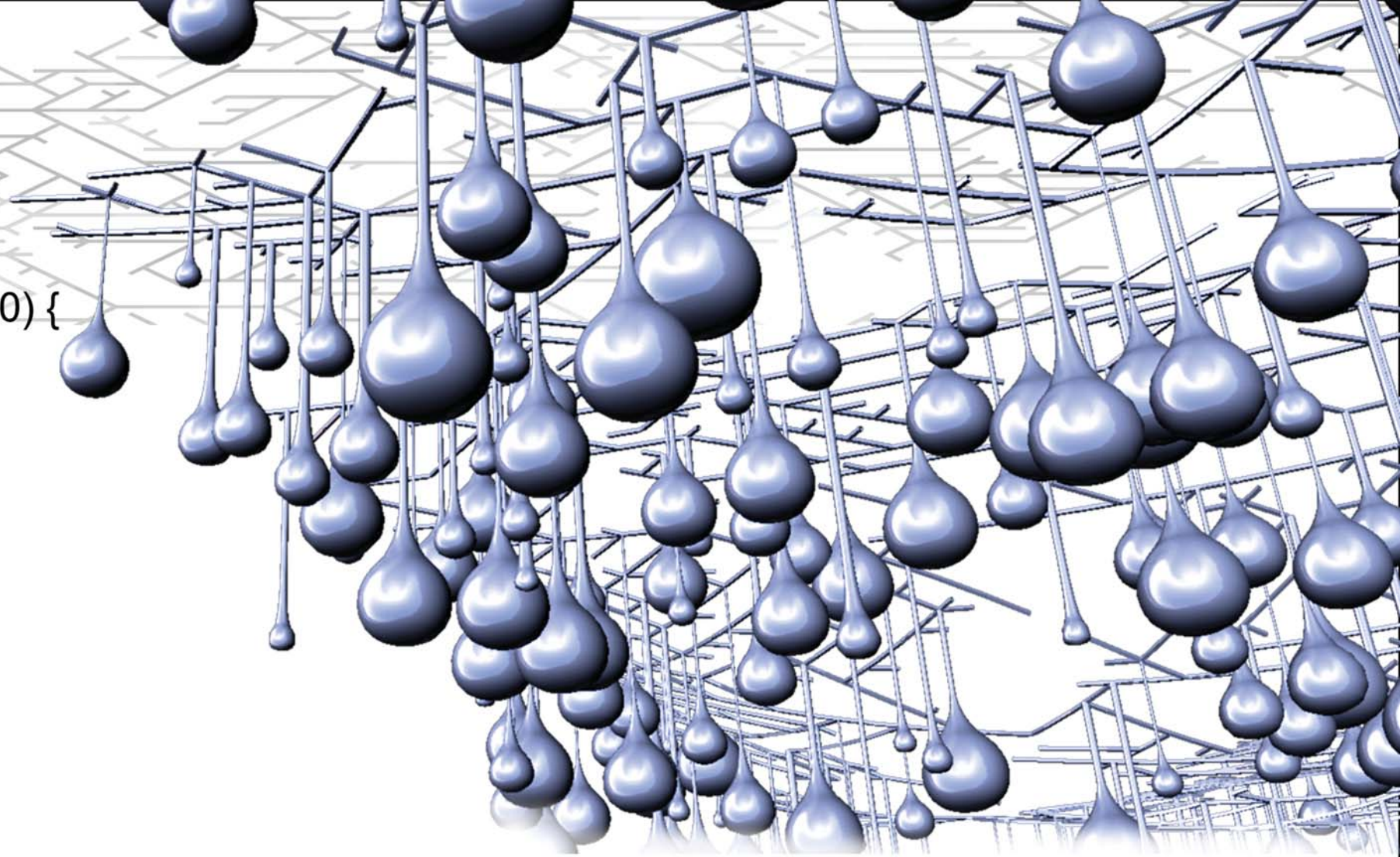
    LineAgent(IVec pt, IVec dir, ISurface s) {
        pt1 = pt;
        pt2 = pt.dup().add(dir.dup().len(length));
        surf = s;
    }
}
```

3) Introducing the void interact

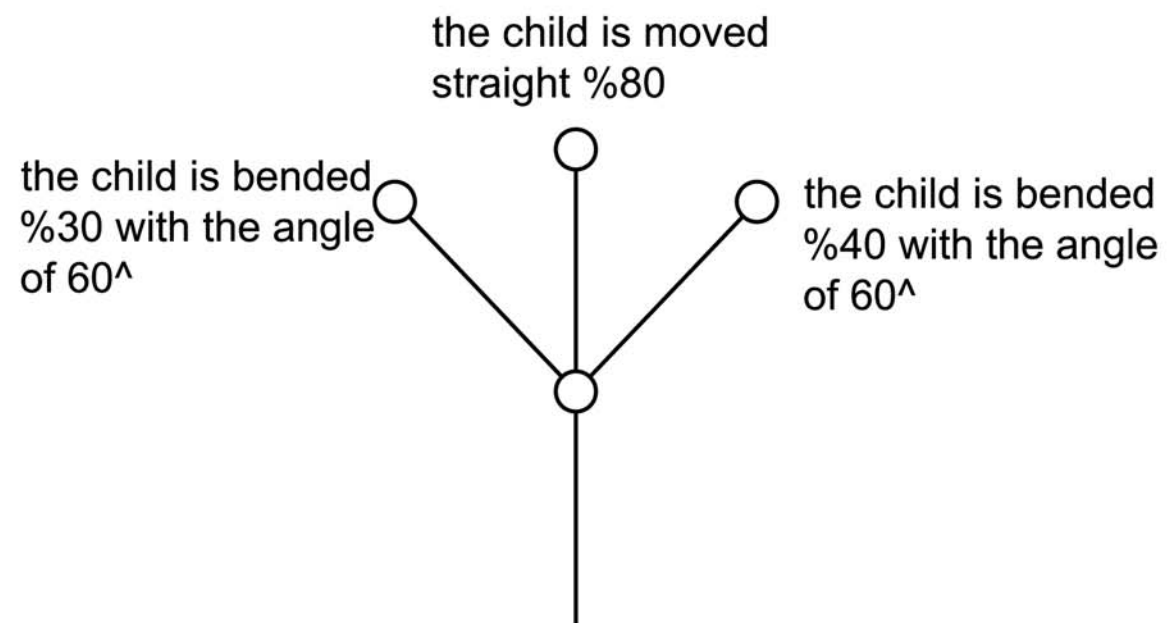
```
void interact(IDynamics agent) {
    if (time == 0) { //only in the first time
        if (agent instanceof LineAgent) {
            LineAgent lineAgent = (LineAgent)agent;
            // checking clearance of end point
            if (lineAgent.pt2.dist(pt2) < clearance) {
                isColliding=true;
            }
        }
    }
}
```

4) Introducing the void update

```
void update() {  
  if (pt2.x < 0.0 || pt2.x > 1.0 || pt2.y < 0.0 || pt2.y > 1.0) {  
    isColliding = true;  
  }  
  
  if (isColliding) {  
    del();  
  }  
  
  else if (time == 0) { //if not colliding  
    IVec surfPt1 = surf.pt(pt1.x, pt1.y);  
    IVec surfPt2 = surf.pt(pt2.x, pt2.y);  
  }  
}
```



5) Introducing the child by percentage

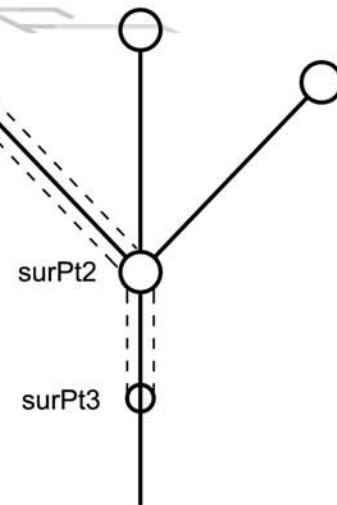


```
if (IRandom.percent(40)) { //bend  
  new LineAgent(pt2, dir.dup().rot(IG.zaxis, PI/3), surf).clr(gray);  
  
  if (IRandom.percent(30)) { //bend the other way  
    new LineAgent(pt2, dir.dup().rot(IG.zaxis, -PI/3), surf).clr(gray);  
  
  if (IRandom.percent(80)) { //straight  
    new LineAgent(pt2, dir.dup(), surf).clr(gray);  
  }  
}
```

5) Drawing the pipes

the child is moved
straight %80

the child is bended
%30 with the angle
of 60^

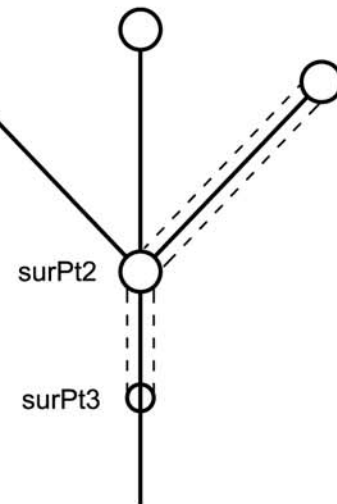


the child is bended
%40 with the angle
of 60^

```
IVec midUV = pt2.cp().add(dir.dup().rot(IG.zaxis, PI/3).div(2));
IVec surfPt3 = surf.pt(midUV.x, midUV.y);
IVec[] cps = new IVec[3];
cps[0] = surfPt1.mid(surfPt2);
cps[1] = surfPt2;
cps[2] = surfPt3;
ICurve amir= new ICurve(cps);
IG.pipe(amir, .02).clr(clr());
```

the child is moved
straight %80

the child is bended
%30 with the angle
of 60^

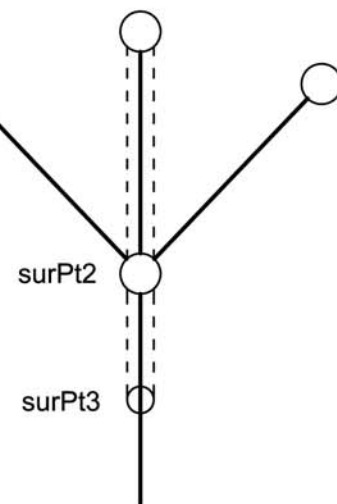


the child is bended
%40 with the angle
of 60^

```
IVec midUV = pt2.cp().add(dir.dup().rot(IG.zaxis, PI/3).div(2));
IVec surfPt3 = surf.pt(midUV.x, midUV.y);
IVec[] cps = new IVec[3];
cps[0] = surfPt1.mid(surfPt2);
cps[1] = surfPt2;
cps[2] = surfPt3;
ICurve amir= new ICurve(cps);
IG.pipe(amir, .02).clr(clr());
```

the child is moved
straight %80

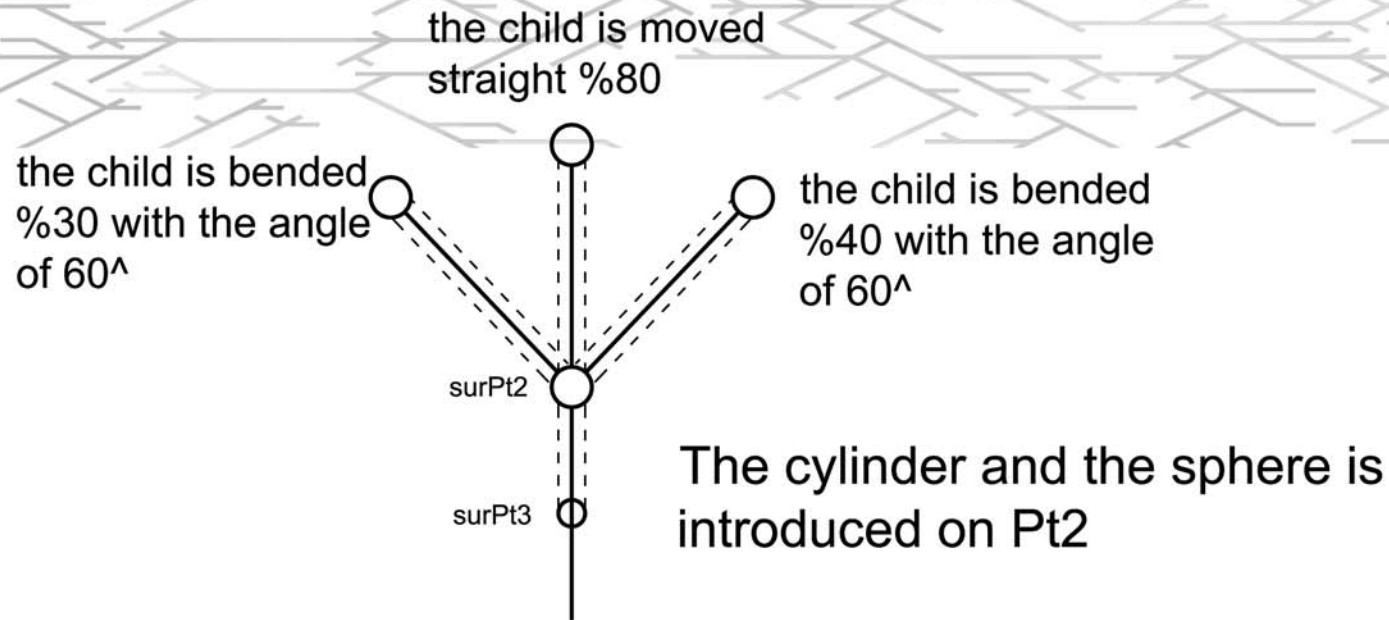
the child is bended
%30 with the angle
of 60^



the child is bended
%40 with the angle
of 60^

```
IVec surfPt3 = surf.pt(midUV.x, midUV.y);
IVec[] cps = new IVec[2];
cps[0] = surfPt1.mid(surfPt2);
cps[1] = surfPt3;
ICurve amir= new ICurve(cps);
IG.pipe(amir, .02).clr(clr());
```

6) Drawing the cylinder and sphere in random height



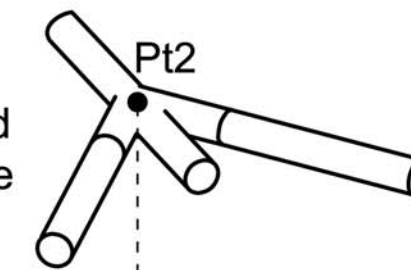
IVec end = pt2;

```
IVec endPt = surf.pt(end.x, end.y);
double offsetDepth = IRandom.get(-0.5, -1);
double radius = IRandom.get(0.01, .02);
double radiusSphere = IRandom.get(.1, 0.2);
IVec endPt1 = surf.pt(end.x, end.y, offsetDepth);
new ICylinder (endPt, endPt1, radius);
new ISphere (endPt1, radiusSphere);
```

6.1) The point Pt2 is copied in Z axis by random number:

```
IVec endPt2 = endPt.cp().add( 0,0,IRandom.get(-1, -3));
```

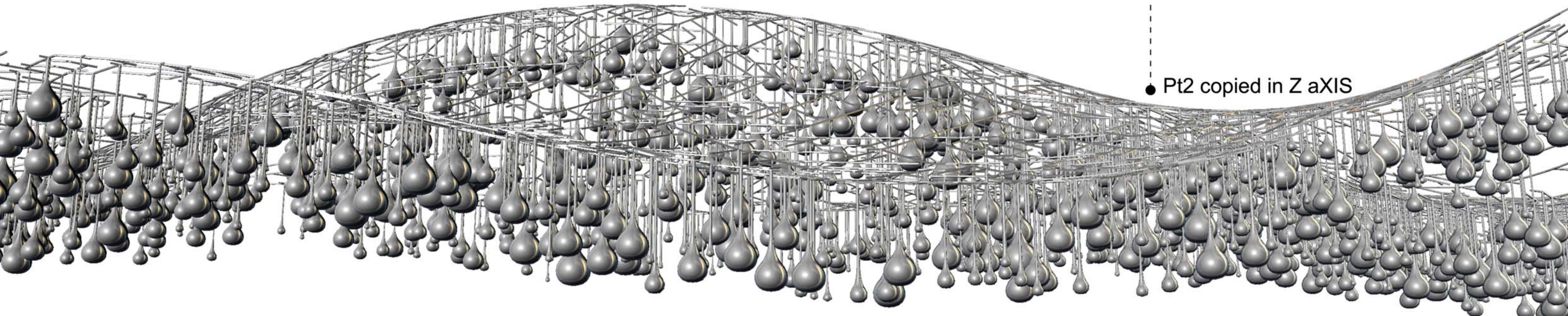
the child is bended %30 with the angle of 60^



the child is bended %40 with the angle of 60^

the child is moved straight %80

● Pt2 copied in Z aXIS



6.2) The copied Pt2 creates the curve line of balloons

ICircle circle1

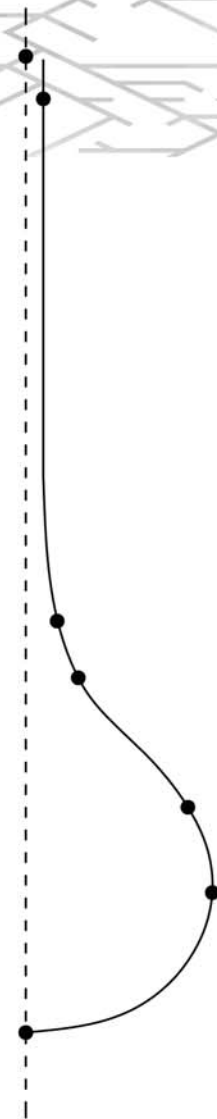
ICircle circle2

ICircle circle3

ICircle circle5

ICircle circle6

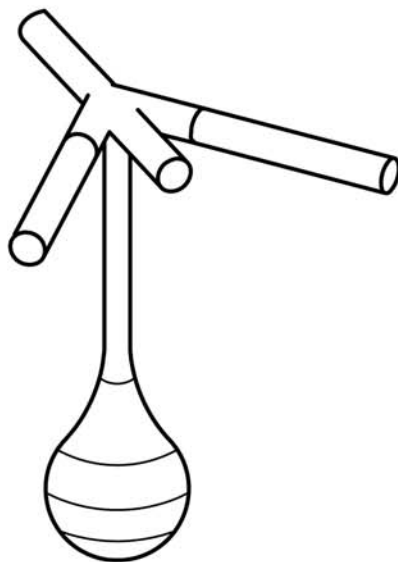
ICircle circle7



```
ISurface balloonSurface(IVec pipeCenter, IVec sphereCenter,
                        double pipeRadius, double sphereRadius){
```

```
    IVec dir = sphereCenter.dif(pipeCenter);
    //ISphere sphere = new ISphere(sphereCenter, sphereRadius);
    IVec center1 = pipeCenter;
    ICircle circle1 = new ICircle(center1, dir, pipeRadius);
    IVec center2 = center1.cp().add(dir.cp().len( (dir.len()-sphereRadius)/2
));
    ICircle circle2 = new ICircle(center2, dir, pipeRadius);
    IVec center3 = center1.cp().add(dir.cp().len(dir.len()-sphereRadius));
    ICircle circle3 = new ICircle(center3, dir, pipeRadius);
    ICircle circle4 = new ICircle(center3, dir, sphereRadius);
    IVec center4 = sphereCenter;
    ICircle circle5 = new ICircle(center4, dir, sphereRadius);
    IVec center5 = center4.cp().add(dir.cp().len(sphereRadius));
    ICircle circle6 = new ICircle(center5, dir, sphereRadius*0.7);
    ICircle circle7 = new ICircle(center5, dir, 0);
```

6.3) Create a loft with the curves



```
ICurve[] circles = new ICurve[6];
```

```
    circles[0] = circle1;
```

```
    circles[1] = circle2;
```

```
    circles[2] = circle3;
```

```
    circles[3] = circle5;
```

```
    circles[4] = circle6;
```

```
    circles[5] = circle7;
```

```
    circle1.del();
```

```
    circle2.del();
```

```
    circle3.del();
```

```
    circle4.del();
```

```
    circle5.del();
```

```
    circle6.del();
```

```
    circle7.del();
```

```
    return IG.loft(circles, 2);
```

