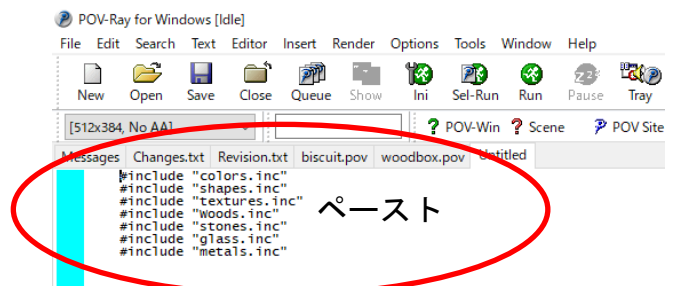
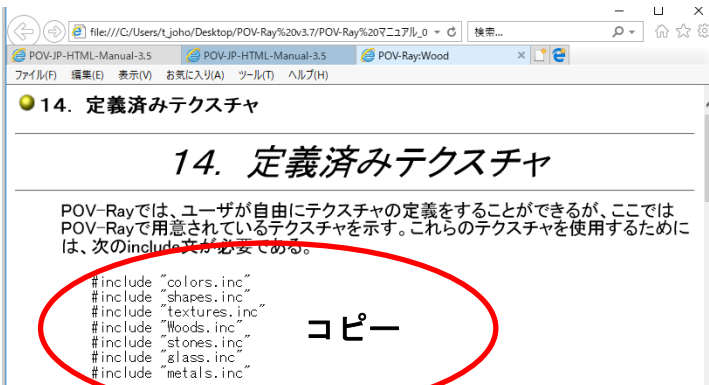
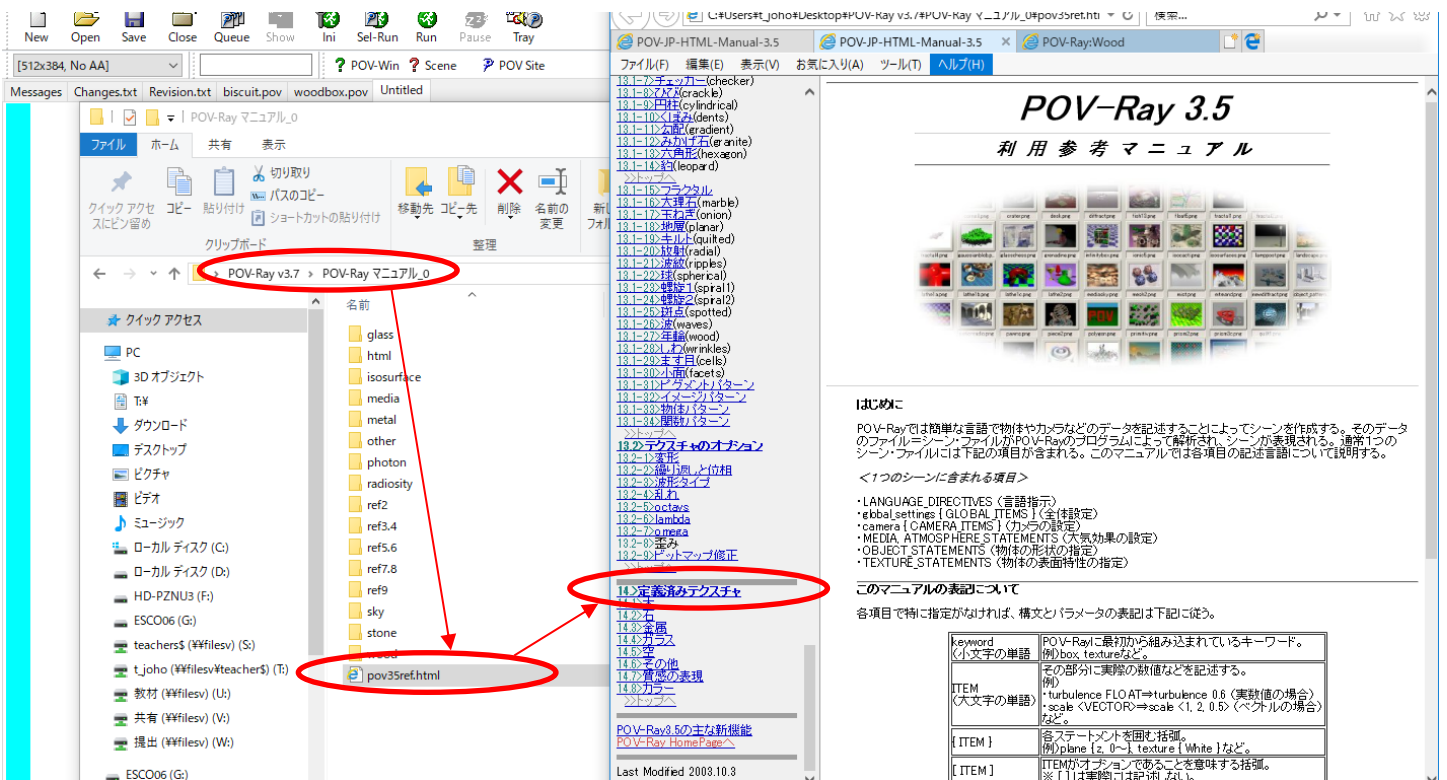


POV-RAY を使った 3D アニメーション概念

目的 : ① プログラミングによる 3DCG デザイン

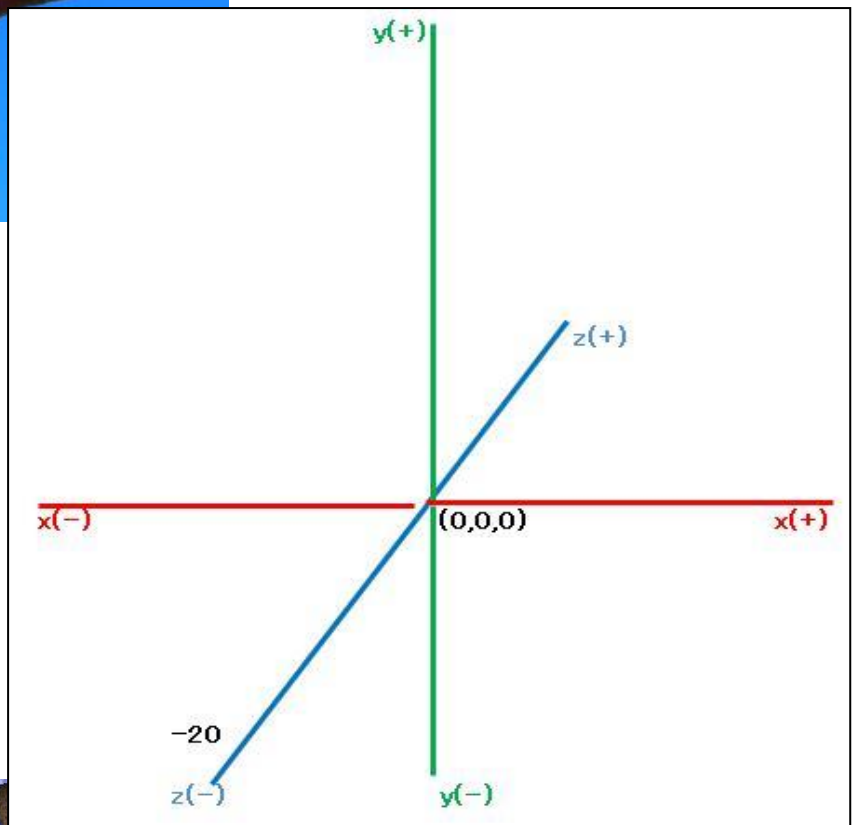
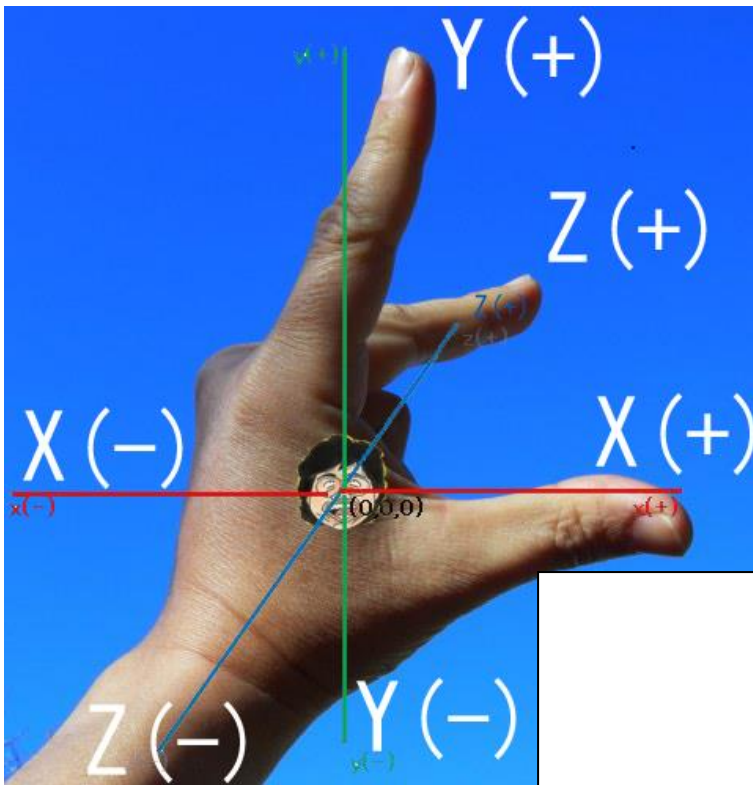
② プログラミングにおける「ライブラリ」活用と手入力の作法

- POV-RAY はレイヤー構造ではなく、個々の object に clock を用いてモーションを設定する。
- 他の MOVIE 作成ソフト素材としてレンダリングする。
- 教材フォルダから Pov-Ray v3.7 をコピーして s7... (T) 以下にペースト
- 作業はデスクトップで実行し、画像は s7... (T) に保存することにします
- Pov-Ray v3.7 フォルダ内の PovRay 実行
- 大分大学工学部建設工学科作成マニュアルから材質定義を埋め込む



開始します ➡ 左手を目の前に出してください

1. POV-RAY 導入



立体表現
Pov-Ray : 左手系 XYZ 座標
(x,y,z)で表記
Camera : 観測者
Light_source : 光源
Object : :物体
光線を追跡して、三者の相関関係で生じる「光」と「影」表現



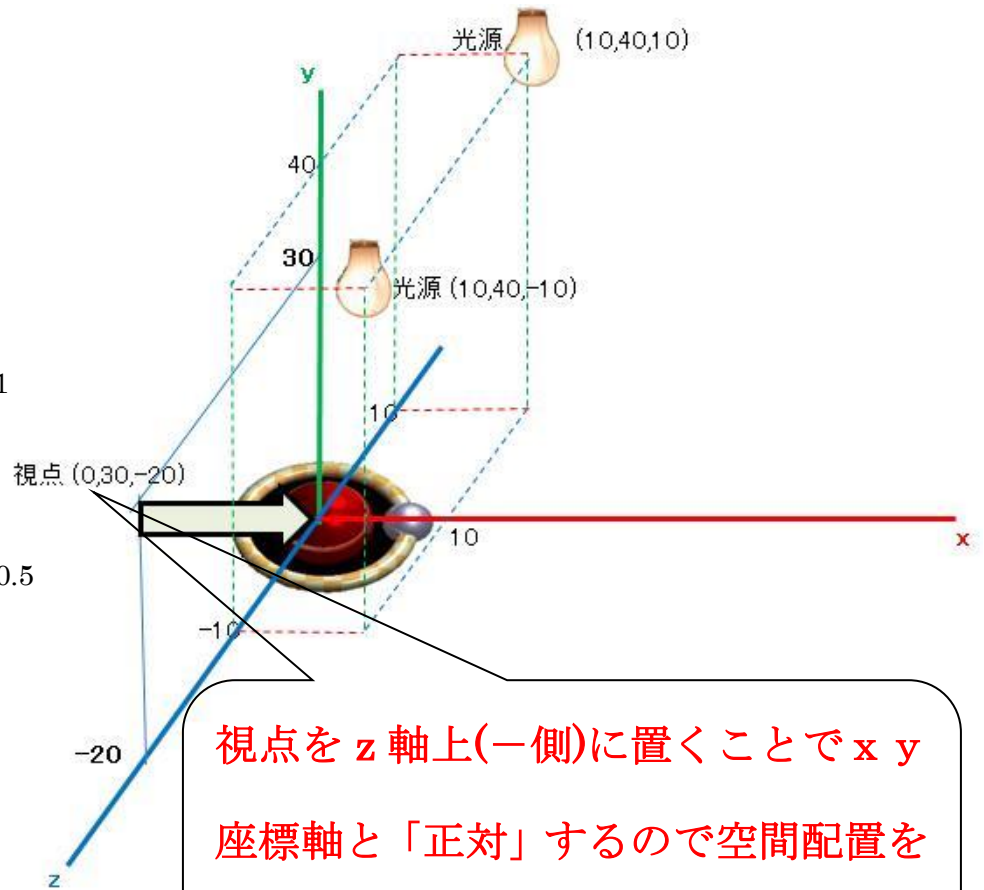
(1) 物体・光源・視点の位置を確認する

```
camera{
location<0,30,-20>
look_at<0,0,0>
angle 30
}

light_source{
<10,40,10> color rgb<1,1,1>*1
}

light_source{
<10,40,-10> color rgb<1,1,1>*0.5
}

//白玉//
object{
sphere<0,0,0>,1
pigment{color rgb<1,1,1>}
finish{phong 0 reflection 0}
translate <0.0,0>
rotate <0,0,0>
} }
```



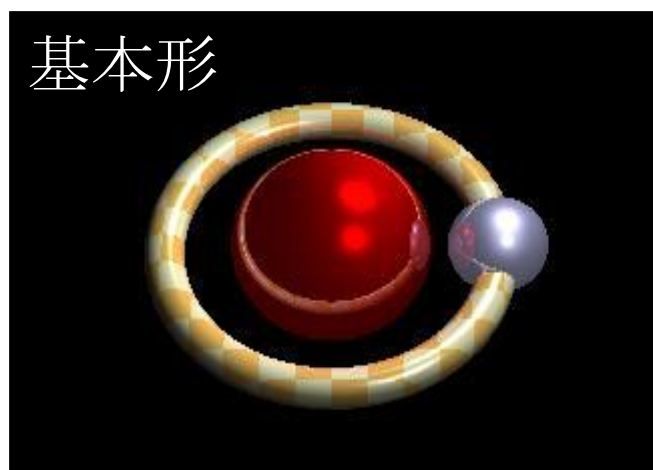
視点を z 軸上(一側)に置くことで x y 座標軸と「正対」するので空間配置を理解しやすい

translate と rotate の順序に注意を払う

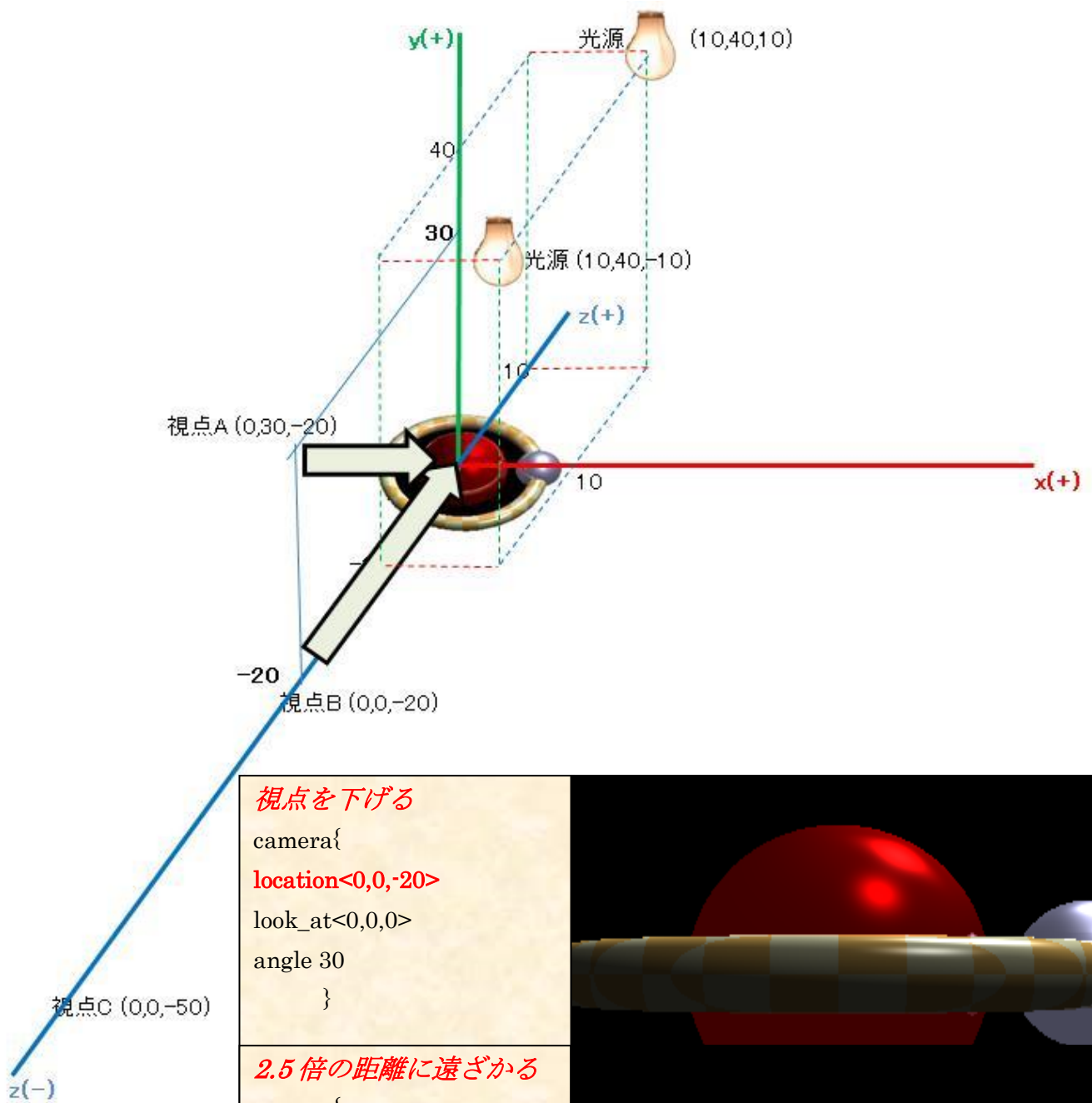


```
//赤玉//
object{
sphere<0,0,0>,3
texture {T_Silver_5C}
pigment{color rgb<1,0,0>}
finish{phong 1 reflection 0.4}
translate <0.0,0>
} }
```

```
//ドーナツ//
torus{5,0.6
pigment{checker color rgb<1,0.8,0.4>*1,
color rgb<1,1,0.8> scale 1}
finish{phong 1 reflection 0.4}
rotate <0,0,>
translate <0,0,0>
} }
```



(2) 視点を変える



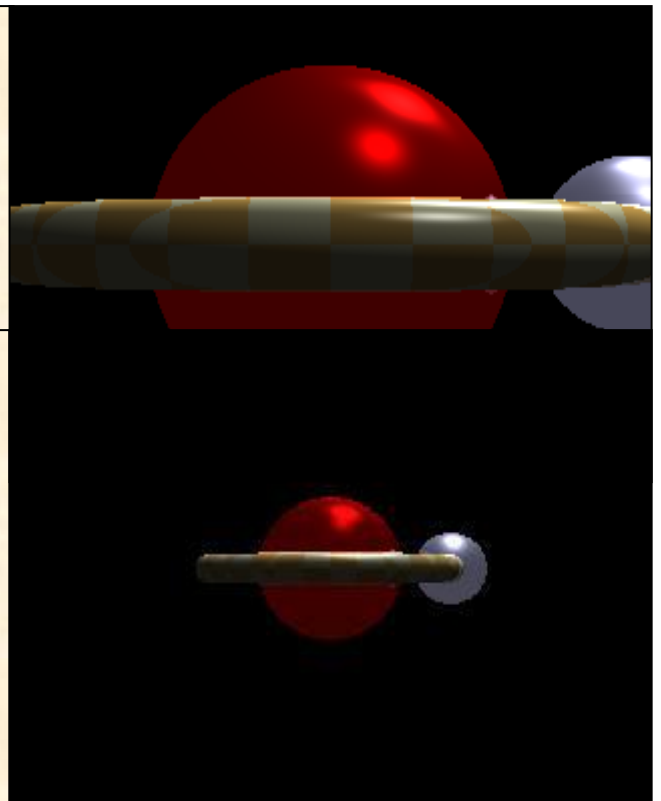
視点を下げる

```
camera{  
  location<0,0,-20>  
  look_at<0,0,0>  
  angle 30  
}
```

2.5 倍の距離に遠ざかる

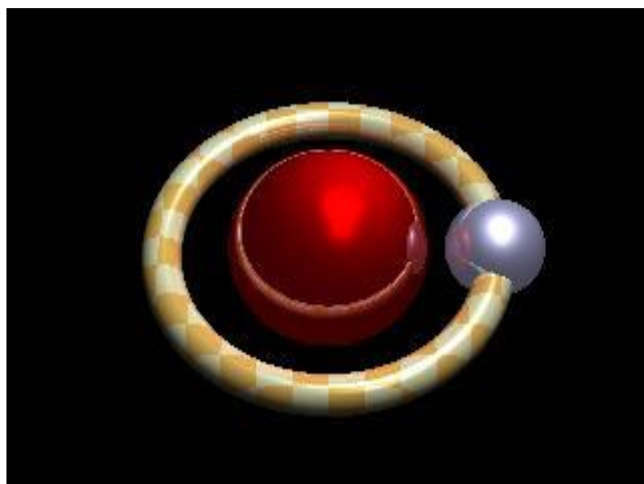
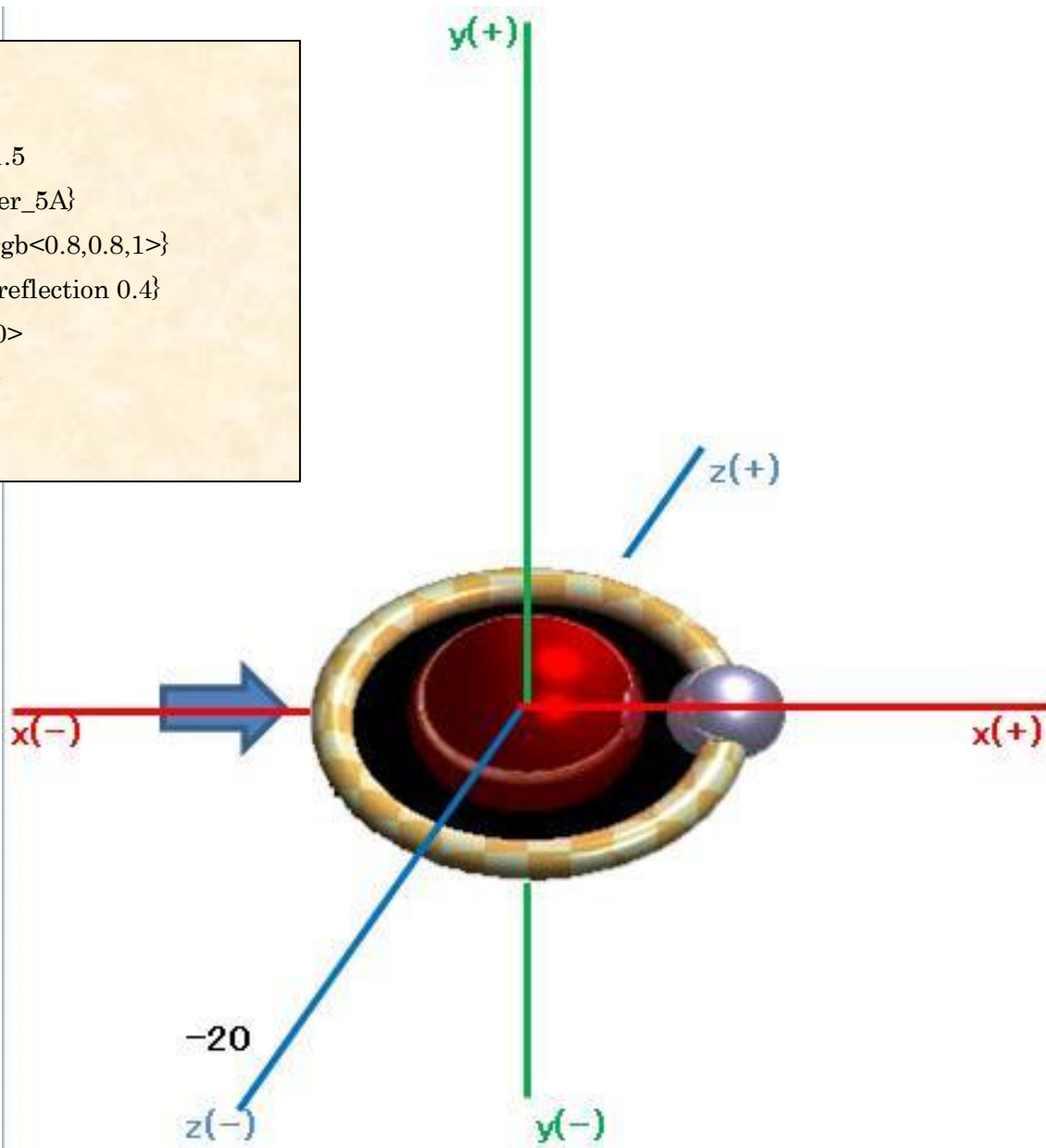
```
camera{  
  location<0,0,-20>*2.5  
  look_at<0,0,0>  
  angle 30  
}
```

視点 C



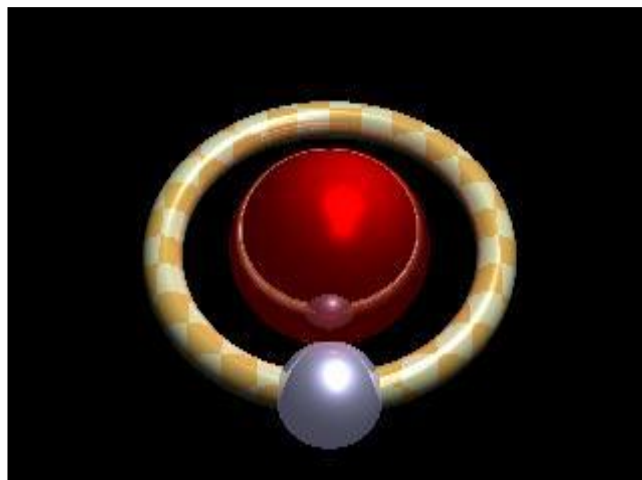
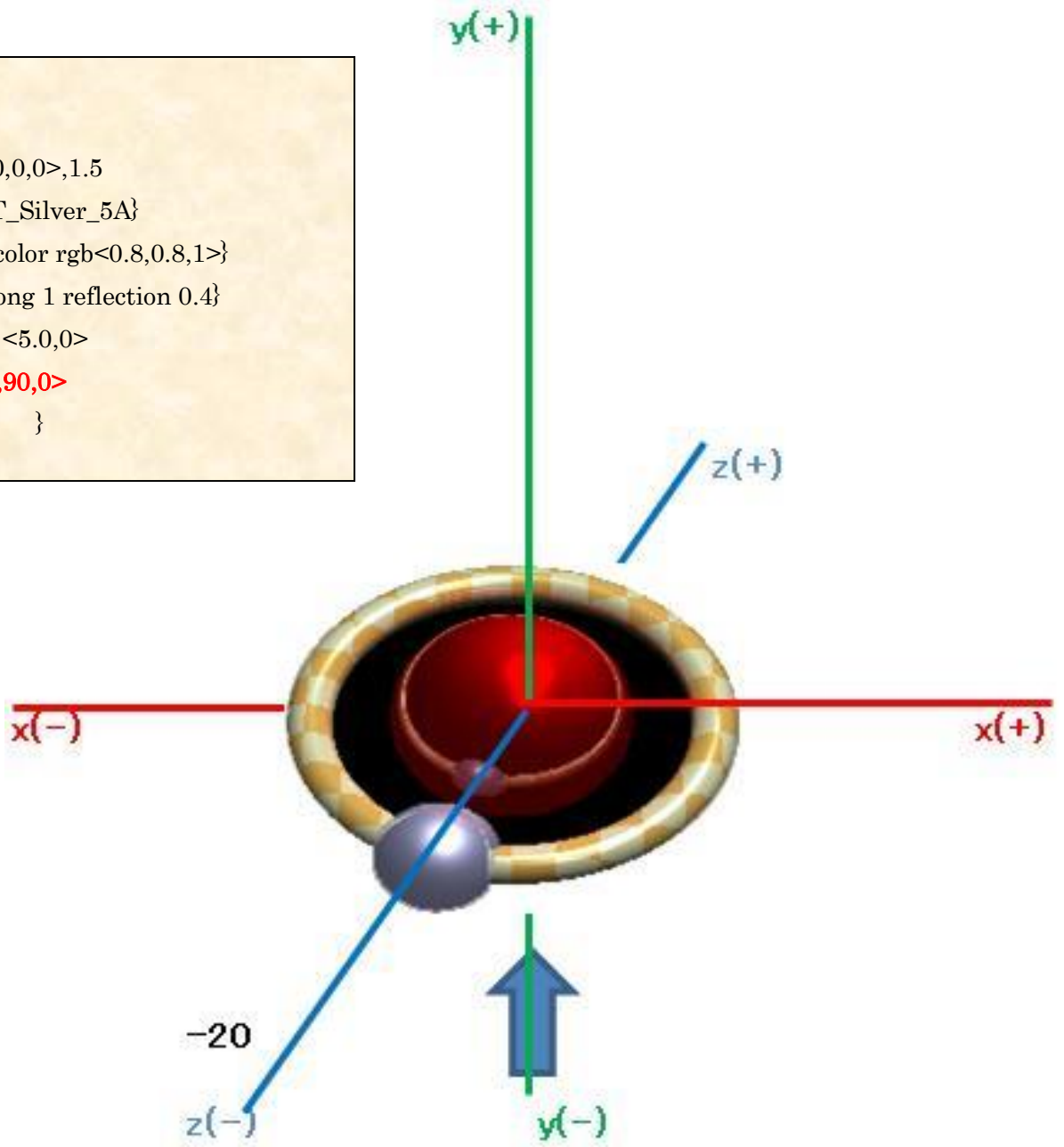
(3) 紫玉をX軸を中心に90°回転する

```
object{  
  sphere{<0,0,0>,1.5  
    texture {T_Silver_5A}  
    pigment{color rgb<0.8,0.8,1>}  
    finish{phong 1 reflection 0.4}  
    translate <5.0,0>  
    rotate <90,0,0>  
  }  
}
```



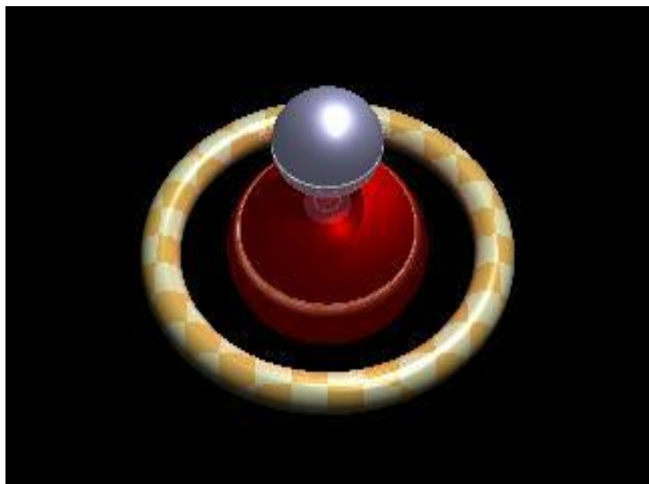
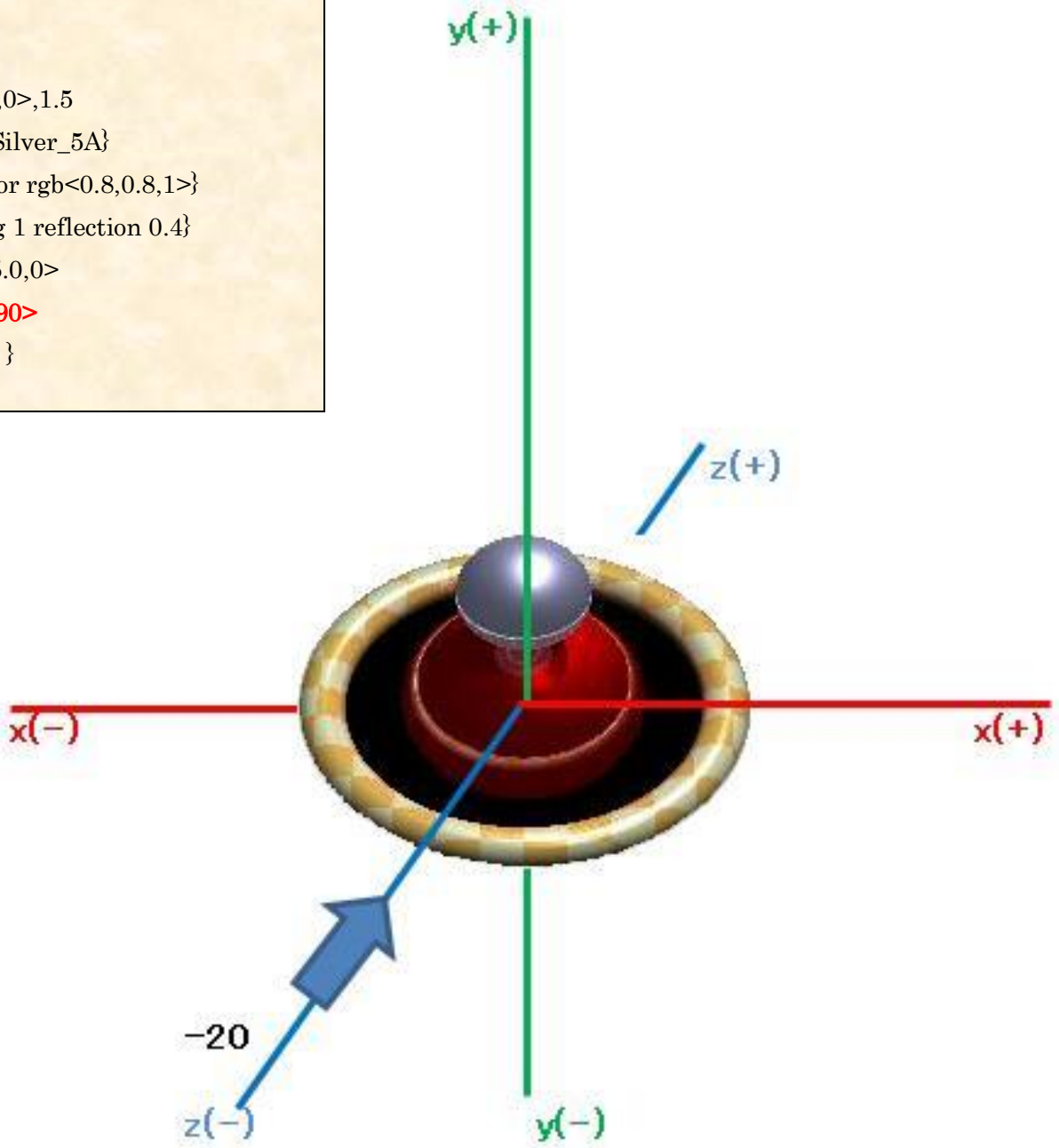
(4) 紫玉をY軸を中心に 90° 回転する

```
object{
  sphere{<0,0,0>,1.5
  texture {T_Silver_5A}
  pigment{color rgb<0.8,0.8,1>}
  finish{phong 1 reflection 0.4}
  translate <5.0,0>
  rotate <0,90,0>
}
```



(5) 紫玉をZ軸を中心に90°回転する

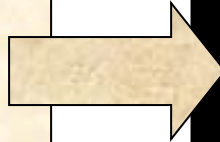
```
object{
  sphere<0,0,0>,1.5
  texture {T_Silver_5A}
  pigment{color rgb<0.8,0.8,1>}
  finish{phong 1 reflection 0.4}
  translate <5.0,0>
  rotate <0,0,90>
}
```



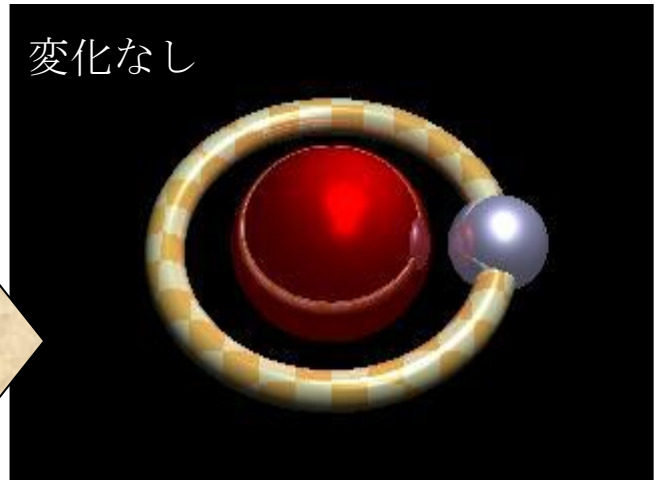
(6) **translate** と **rotate** の順序を入れ替え、紫玉をY軸を中心に 90° 回転する

translate と **rotate** の順序入れ替え

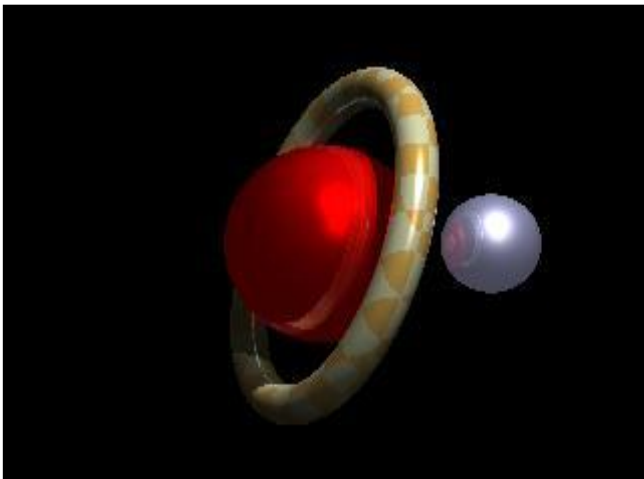
```
object{  
  sphere{<0,0,0>,1.5  
  texture {T_Silver_5A}  
  pigment{color rgb<0.8,0.8,1>}  
  finish{phong 1 reflection 0.4}  
  rotate <0,90,0>  
  translate <5.0,0>  
  }  
}
```



変化なし



理由は？



文字の挿入

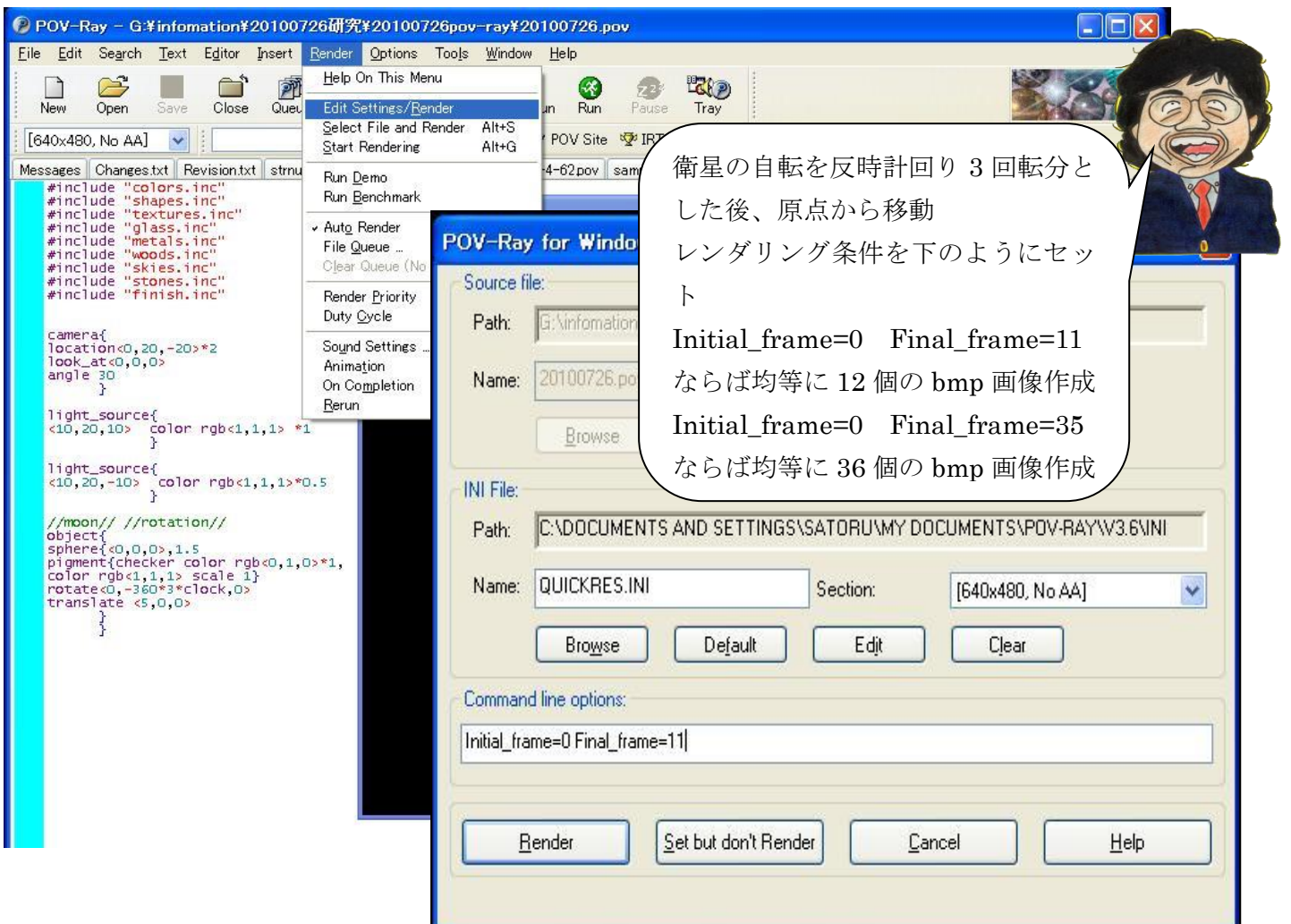
```
text{ttf
"c:¥WINDOWS¥FONTS¥Arial.ttf","strnun
mountain view",2,x*-0.03
scale 3
pigment{color rgb<0.5,1,0.6>*2}
finish{phong 1 reflection 0.2}
rotate <0,0,0>
translate <-20,-5,-20>
}
```

写真(.png) の貼り付け

```
torus { 7,3
pigment{
image_map { png "self_10.png"
map_type 1
interpolate 2 }
rotate <0,-90,0> }
rotate <-20,0,0>
translate <15,-5,0> }
```



2. POV-RAYで3Dアニメーション、3Dムービー素材作成



POV-Ray - G:\information\20100726研究\20100726pov-ray\20100726.pov

File Edit Search Text Editor Insert Render Options Tools Window Help

[640x480, No AA] ? POV-Win ? Scene POV Site IRTC Site

Messages Changes.txt Revision.txt strnun-4-66.pov strnun-4-65.pov strnun-4-62.pov sample.pov sample2.pov sample.pov 20100726.pov

```
#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"
#include "glass.inc"
#include "metals.inc"
#include "woods.inc"
#include "skies.inc"
#include "stones.inc"
#include "finish.inc"

camera{
location<0,20,-20>*2
look_at<0,0,0>
angle 30
}

light_source{
<10,20,10> color rgb<1,1,1> *1
}

light_source{
<10,20,-10> color rgb<1,1,1>*0.5
}

//moon// //rotation//
object{
sphere<0,0,0>,1.5
pigment{checker color rgb<0,1,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,-360*clock,0>
translate <5,0,0>
}

//planet// //rotation//
object{
sphere<0,0,0>,1.5
pigment{checker color rgb<1,0,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,360*2*clock,0>
translate <0,0,0>
}
```

544,2

360*clock 1回転
Initial_frame=0 Final_frame=11
ならば1コマ毎の回転角 $\theta = 30^\circ$
720*clock 2回転
Initial_frame=0 Final_frame=35
ならば1コマ毎の回転角 $\theta = 20^\circ$

POV-Ray - G:\information\20100726研究\20100726pov-ray\20100726.pov

File Edit Search Text Editor Insert Render Options Tools Window Help

[640x480, No AA] initial_frame=0 ? POV-Win ? Scene POV Site IRTC Site

Messages Changes.txt Revision.txt strnun-4-66.pov strnun-4-65.pov strnun-4-62.pov sample.pov sample2.pov sample.pov 20100726.pov

```
#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"
#include "glass.inc"
#include "woods.inc"
#include "skies.inc"
#include "stones.inc"
#include "finish.inc"

camera{
location<0,20,-20>*2
look_at<0,0,0>
angle 30
}

light_source{
<10,20,10> color rgb<1,1,1> *1
}

light_source{
<10,20,-10> color rgb<1,1,1>*0.5
}

//couple two things,translate// //rev//
//moon// //rotation// //revolution//
union{
object{
sphere<0,0,0>,1.5
pigment{checker color rgb<0,1,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,-360*clock,0>
translate <5,0,0>
rotate<0,-360*clock,0>
}
}

//planet// //rotation//
object{
sphere<0,0,0>,1.5
pigment{checker color rgb<1,0,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,360*clock,0>
translate <0,0,0>
}

translate <10,0,0>
rotate<0,-360*clock,0>

//star//
object{
sphere<0,0,0>,1.5
pigment{checker color rgb<0,0,1>*1,
color rgb<1,1,1> scale 1}
rotate<0,-360*clock,0>
translate <0,0,0>
}
```

627,317

衛星を惑星の周りで公転させ、Union{ }で二つのobjectを結合させる。その上で移動させた後、原点に恒星を作る。

POV-Ray - G:\information\20100726研究\20100726pov-ray\20100726.pov

File Edit Search Text Editor Insert Render Options Tools Window Help

New Open Save Close Queue Rerun Hide Ini Sel-Run Run Pause Tray

[640x480, No AA] initial_frame=0 ? POV-Win ? Scene ? POV Site ? IRTC Site

Messages Changes.txt Revision.txt strnun-4-66.pov strnun-4-65.pov strnun-4-62.pov sample.pov sample2.pov sample.pov 20100726.pov

```
#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"
#include "glass.inc"
#include "metals.inc"
#include "woods.inc"
#include "skies.inc"
#include "stones.inc"
#include "finish.inc"

camera{
location<10,20,-20>*2
look_at<0,0,0>
angle 30
}

light_source{
<10,20,10> color rgb<1,1,1> *1
}

light_source{
<10,20,-10> color rgb<1,1,1>*0.5
}

//All combination// //Movement//
//couple two things,translate// //revo
//moon// //rotation// //revolution//
union{
union{
object{
sphere{<0,0,0>,1.5
pigment{checker color rgb<0,1,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,-360*3*clock,0>
translate <5,0,0>
rotate<0,-360*clock,0>
}
}

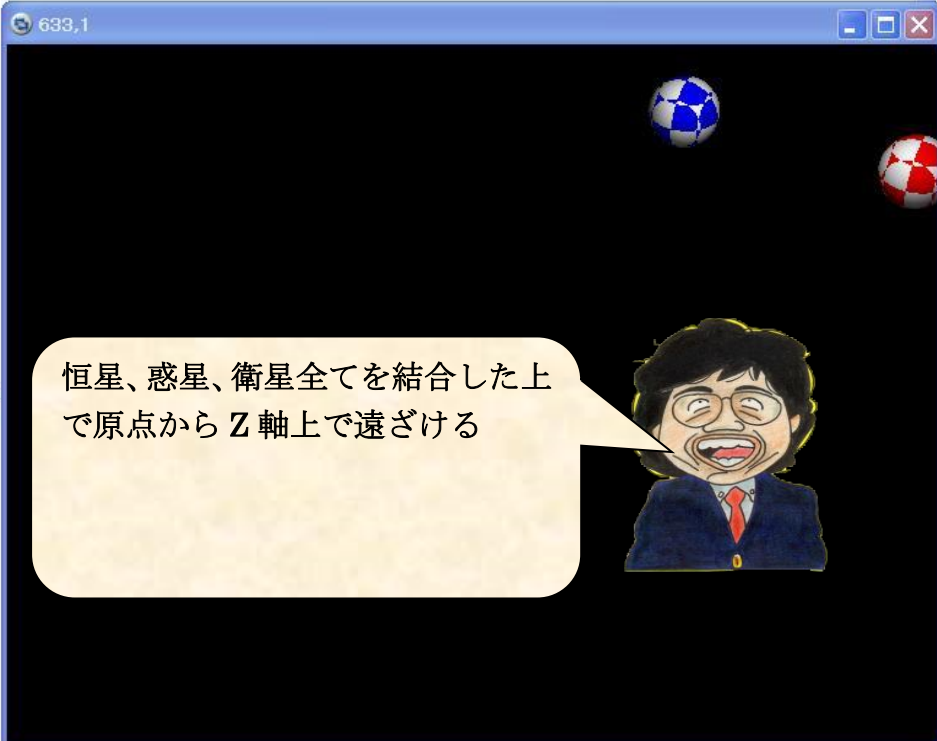
//planet// //rotation//
object{
sphere{<0,0,0>,1.5
pigment{checker color rgb<1,0,0>*1,
color rgb<1,1,1> scale 1}
rotate<0,360*2*clock,0>
translate <0,0,0>
}
}

translate <10,0,0>
rotate<0,-360*clock,0>
}

//star//
object{
sphere{<0,0,0>,1.5
pigment{checker color rgb<0,0,1>*1,
color rgb<1,1,1> scale 1}
rotate<0,-360*clock,0>
translate <0,0,0>
}
}

translate <0,0,20*clock>
}
}
```

633.1



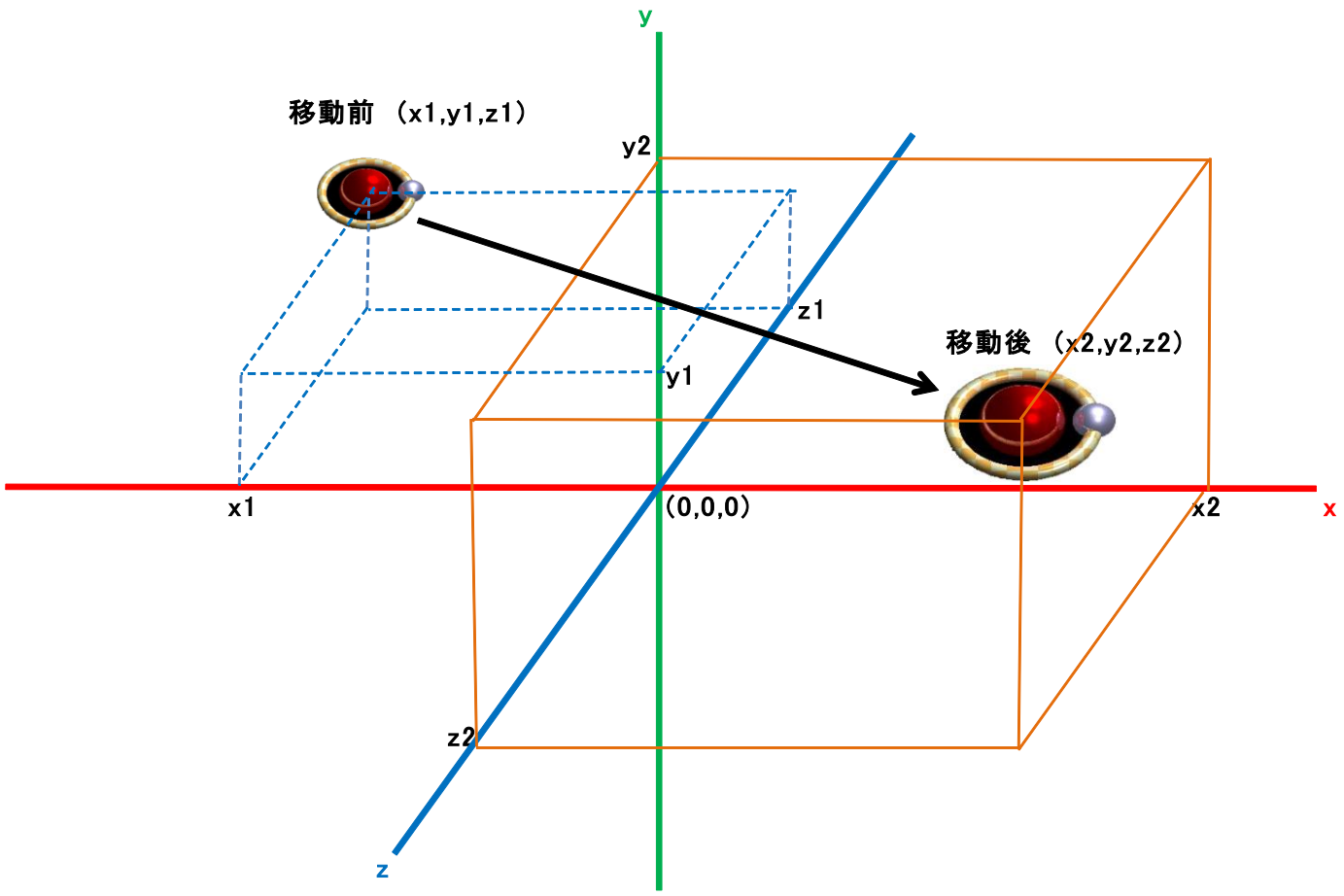
恒星、惑星、衛星全てを結合した上で原点からZ軸上で遠ざける

自転は原点(0,0,0)にて作成すべし！

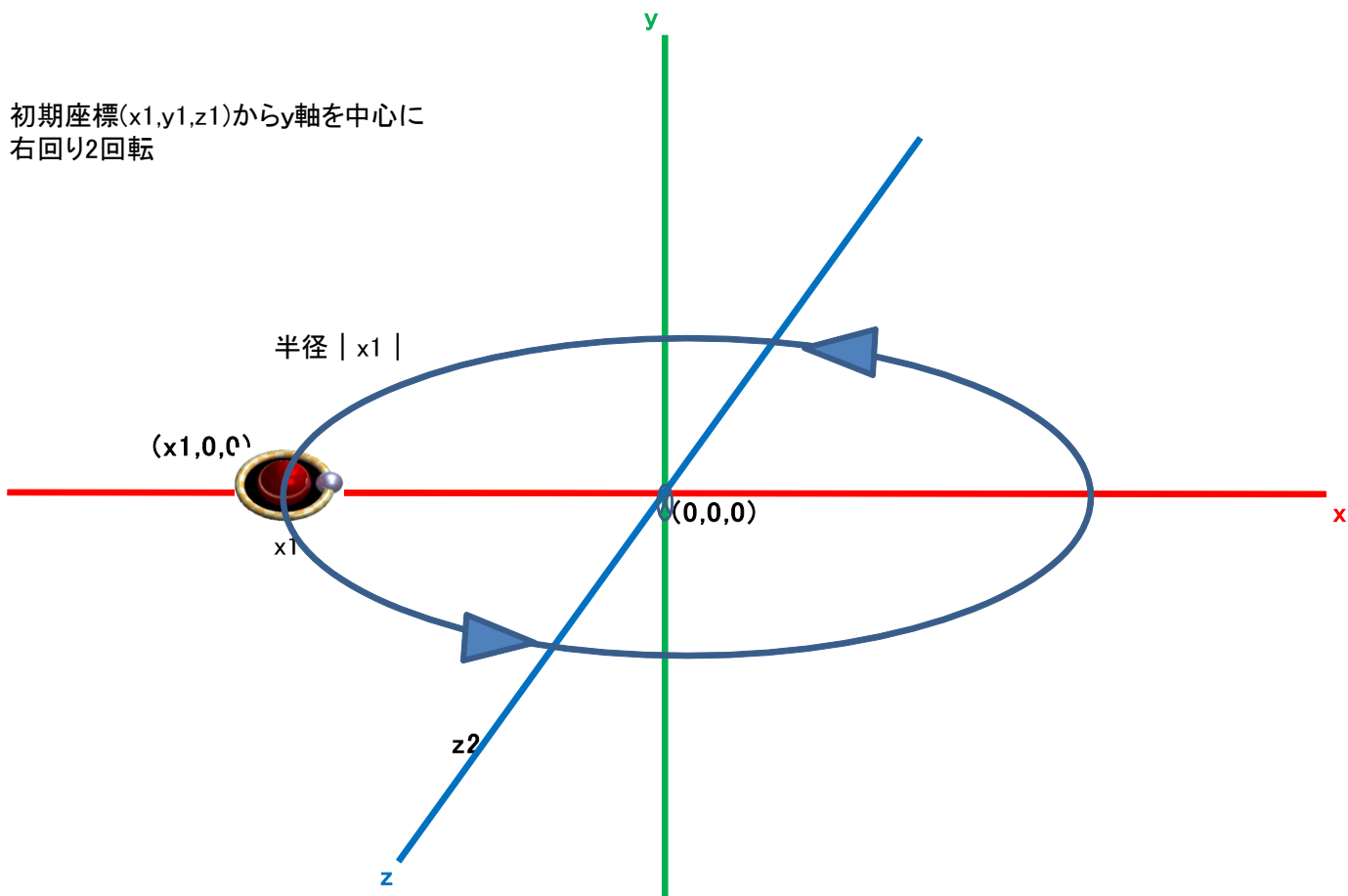
公転は translate を先に用いて原点(0,0,0)以外に移動し作成すべし！

…です。

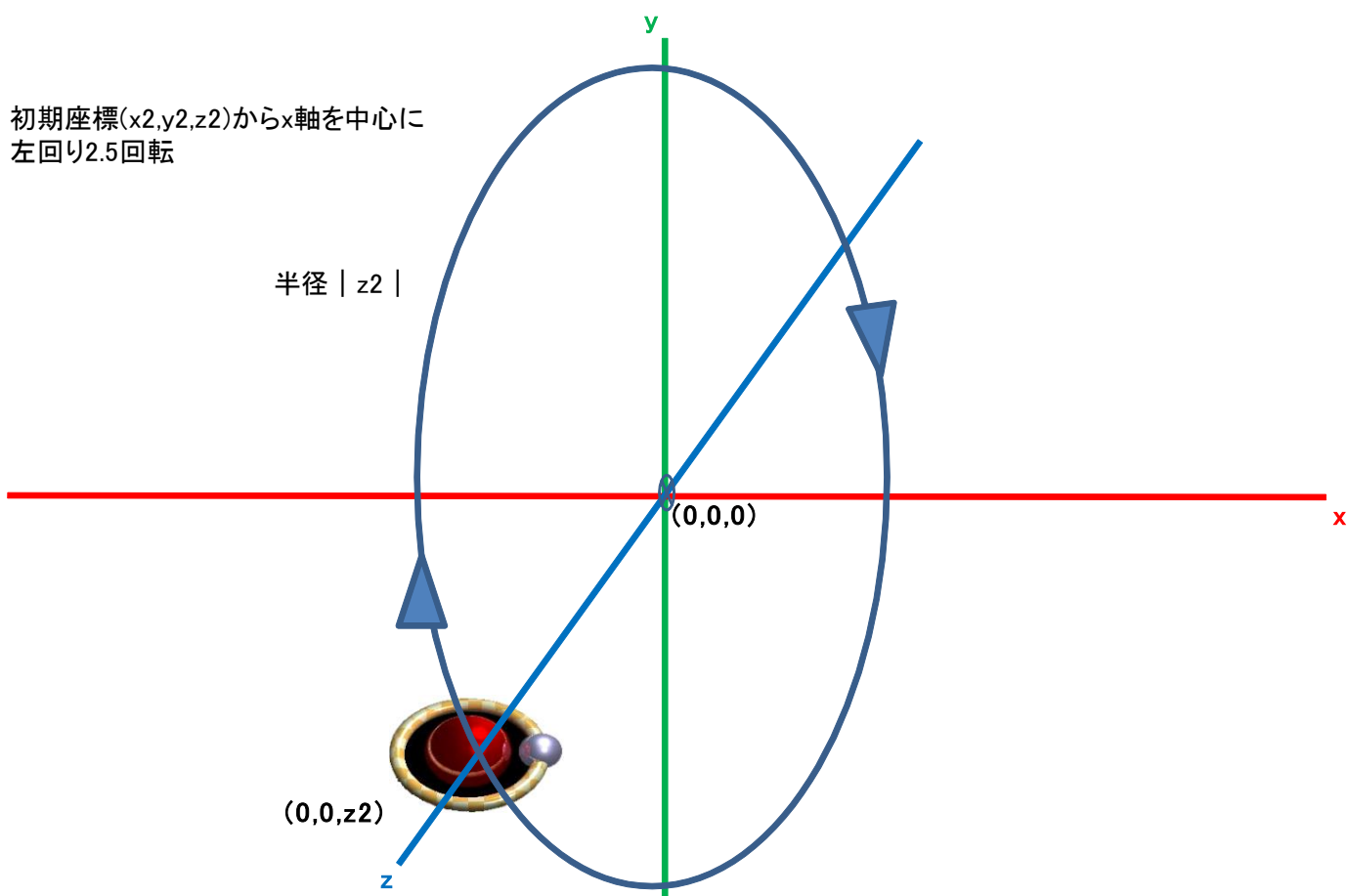




初期座標 (x_1, y_1, z_1) からy軸を中心に
右回り2回転




初期座標 (x_2, y_2, z_2) からx軸を中心に
左回り2.5回転



PCを開き、Pov-Rayを起動

C:\Program Files\POV-Ray\v3.7\binへ進み

 pvengine64.exe 2013/09/24 20:18 アプリケーション 6,974 KB を開く

定義追加

```
#include "stars.inc"
```

```
#include "skies.inc"
```

その他のオブジェクト

箱

```
box{<-2,-2,-2><2,2,2>
```

```
texture {pigment { DMFWood1} }
```

```
rotate<0,60,0>
```

```
translate<0,0,3>
```

```
}
```

板 (無限)

```
plane{y,-8
```

```
texture{T_Stone2}
```

```
}
```

円錐

```
cone{<0,3,0>,0,<0,0,0>,2
```

```
texture {Copper_Texture}
```

```
translate<0,-2,0>
```

```
}
```

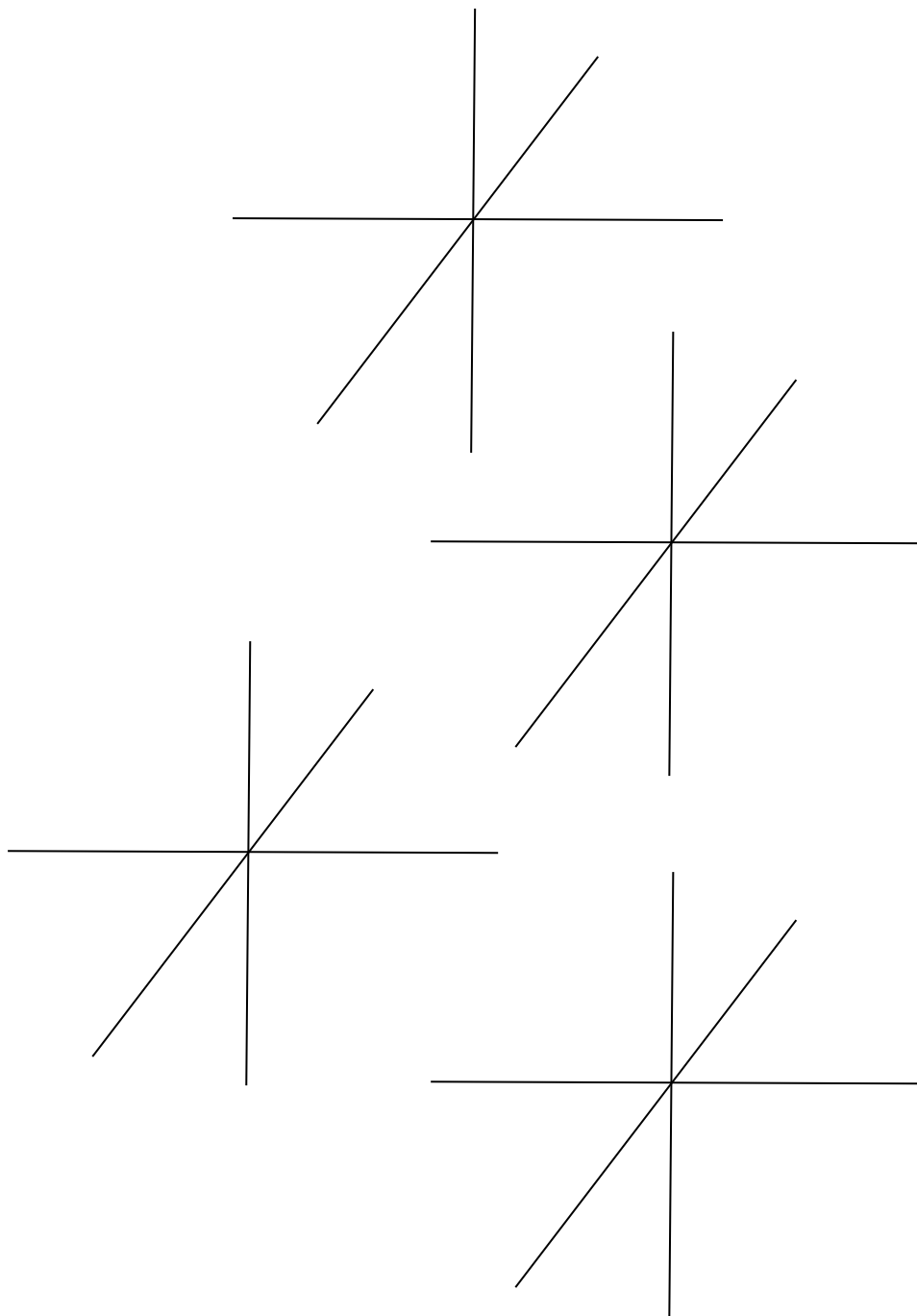
円柱

```
cylinder {<0,3,0>,<0,0,0>,2
```

```
texture {Copper_Texture}
```

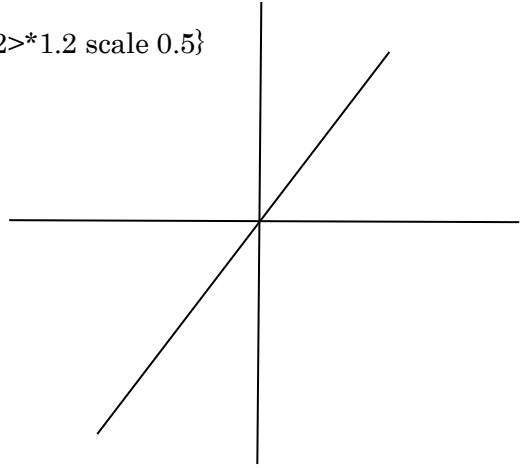
```
translate<0,-4,0>
```

```
}
```



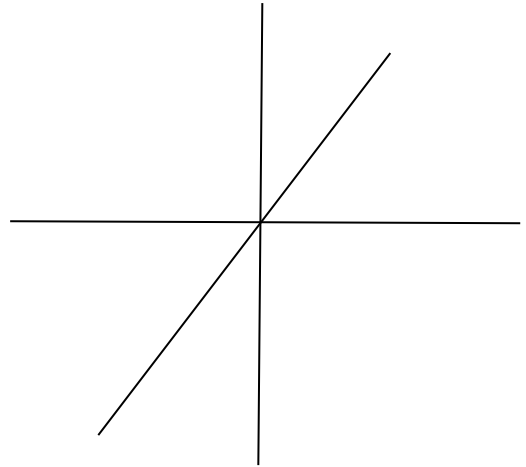
三角形

```
triangle{<-1,-1,0>,<1,-1,0>,<0,2,0>  
  pigment{checker color rgb<1,1,1>*1.2, color rgb<1,0.2,0.2>*1.2 scale 0.5}  
  finish{phong 0 reflection 0.2}  
  rotate <0,0,0>  
  translate <0,0,-3>  
}
```



スポットライト

```
spotlight  
  point_at <0,1,0>  
  radius 5  
  falloff 45  
  tightness 0  
}
```

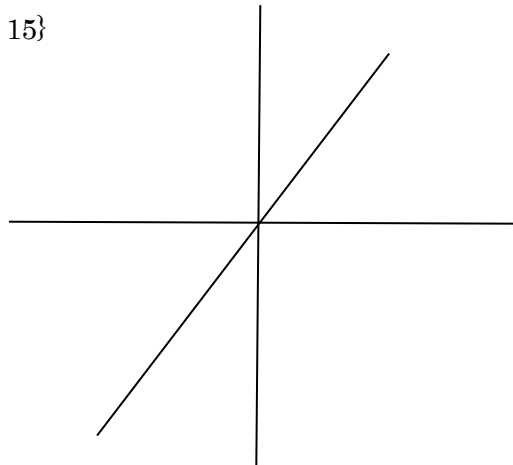


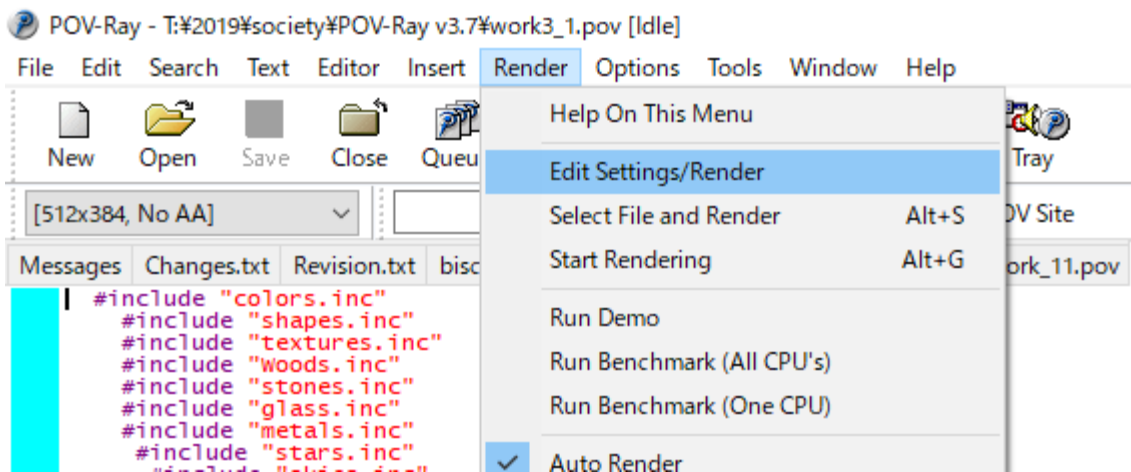
雲

```
sky_sphere{ S_Cloud1}  
sky_sphere{ S_Cloud2}  
sky_sphere{ S_Cloud3}  
sky_sphere{ S_Cloud4}
```

夜空

```
sphere { <0,0,0>,5 inverse texture { Starfield1 } scale 15}  
sphere { <0,0,0>,5 inverse texture { Starfield2 } scale 15}  
sphere { <0,0,0>,5 inverse texture { Starfield3 } scale 15}
```





画像貼り付け

```
sphere {<0 , 0 , 0> , 1.}
```

```
texture {
```

```
  pigment {
```

```
    image_map {
```

```
      jpeg "earthmap.jpg" map_type 1
```

```
    }
```

```
  }
```

```
}
```

```
rotate <0,-360*clock,0>
```

```
finish {
```

```
  ambient 0
```

```
}
```