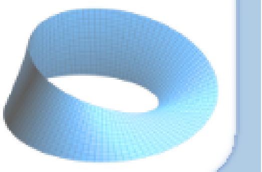
module

B1
Mathematics

subject

104intersection
Third dimnesion and quadratic equations



Rendering formalities

binary name: 104intersection
repository name: 104intersection
repository rights: ramassage-tek
language: C, C++, perl 5, python 2.7, python 3.4, ruby 2.1, php 5.6, bash 4
compilation *(if necessary)*: via Makefile, includig re, clean and fclean rules
group size: 1 to 2

Your repository shall contain the totality of your sources, but no useless file (binary, test files, obj files,...).

Subject

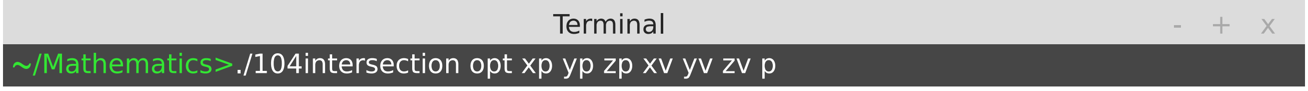
To code a software to create synthesis images (it may remind you something....), potential intersection points between light rays and scene objects (here cylinders, spheres and cones) must be computed.
This is exactly what you have to do.

To do so, you need to write a 3 dimensional equation of the considered surface, and inject into it the equation of the straight line representing the light ray. You'll get a quadratic equation, with 0, 1 or 2 solutions, that will give you the intersection points coordinates (one do not take into account the possibility that the equation has an infinite number of solutions here).
The straight line is defined by the coordinates of a point by which it goes through and by the coordinates of a direction vector.

O being the origin of the coordinate system, and X, Y and Z the axis, the surfaces that has to be handled in this project are :

1. O-centered spheres,
2. cylinders of revolution around Z axis,
3. cones of revolution around Z axis which apex is O.

Prototyping



```
~/Mathematics>./104intersection opt xp yp zp xv yv zv p
```

opt number of the option : 1 for a sphere, 2 for a cylinder, 3 for a cone
(xp, yp, zp) coordinates of the point by which the light ray goes through
(xv, yv, zv) coordinates of the direction vector of the light ray
p parameter : radius of the sphere, radius of the cylinder or angle formed by the cone and the *Z*-axis.

> Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Bonus

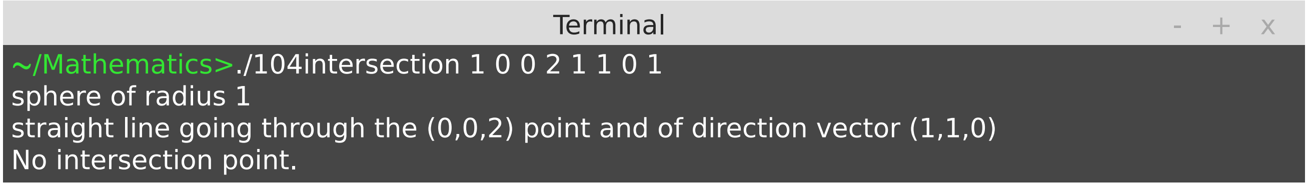
Customize your project adding as many bonus points as possible :

- adding other surfaces,
- managing several simultaneous surfaces,
- ...

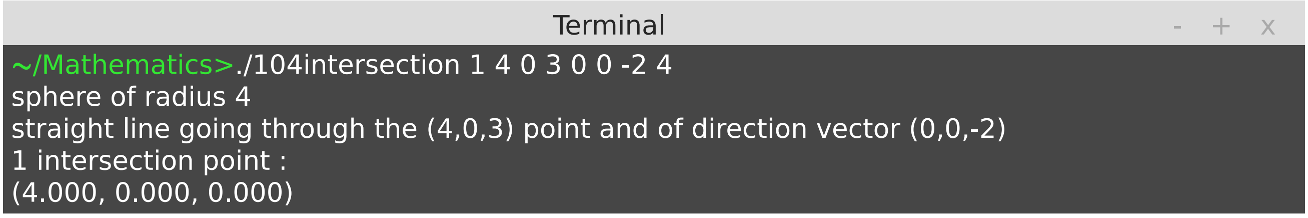
> Let the project with no bonus file in the root directory, and commit your whole project (including bonus and a potential Makefile) in a directory named *bonus*.

Examples

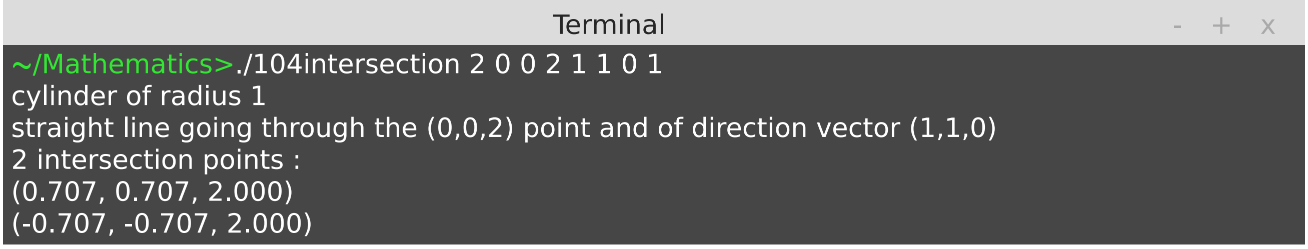
Your program output has to be strictly identical to the one below.



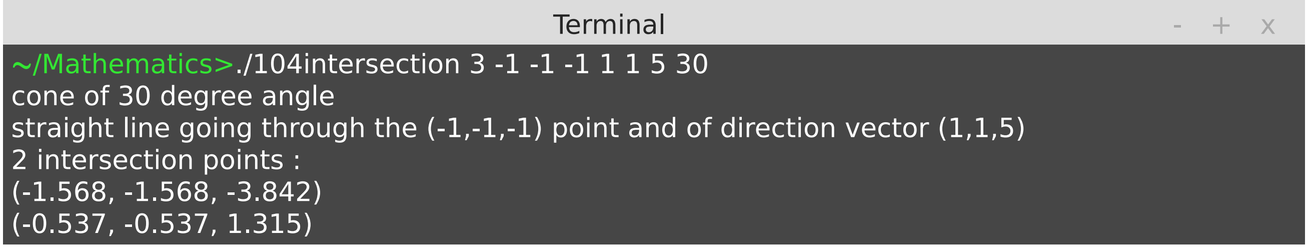
```
~/Mathematics>./104intersection 1 0 0 2 1 1 0 1
sphere of radius 1
straight line going through the (0,0,2) point and of direction vector (1,1,0)
No intersection point.
```



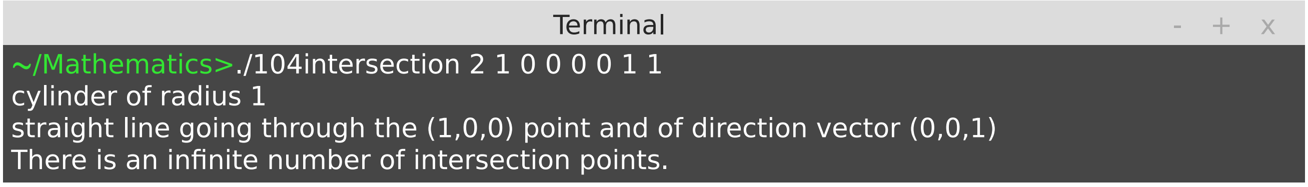
```
~/Mathematics>./104intersection 1 4 0 3 0 0 -2 4
sphere of radius 4
straight line going through the (4,0,3) point and of direction vector (0,0,-2)
1 intersection point :
(4.000, 0.000, 0.000)
```



```
~/Mathematics>./104intersection 2 0 0 2 1 1 0 1
cylinder of radius 1
straight line going through the (0,0,2) point and of direction vector (1,1,0)
2 intersection points :
(0.707, 0.707, 2.000)
(-0.707, -0.707, 2.000)
```



```
~/Mathematics>./104intersection 3 -1 -1 -1 1 1 5 30
cone of 30 degree angle
straight line going through the (-1,-1,-1) point and of direction vector (1,1,5)
2 intersection points :
(-1.568, -1.568, -3.842)
(-0.537, -0.537, 1.315)
```



```
~/Mathematics>./104intersection 2 1 0 0 0 0 1 1
cylinder of radius 1
straight line going through the (1,0,0) point and of direction vector (0,0,1)
There is an infinite number of intersection points.
```