In [154]:
```
from numpy.linalg import eig
from random import random
import numpy as np
from math import acos, pi, sin, cos

def get_unit_vector():
    s = np.array([2*random()-1, 2*random()-1, 2*random()-1])
    s = s/np.linalg.norm(s)
    return s

def get_s_matrix(s):
    return np.array([[0, -s[2], s[1]], [s[2], 0, -s[0]], [-s[1], s[0], 0]])

def get_rotation_matrix(s, phi):
    S = get_s_matrix(s)
    R = np.eye(3) + sin(phi) * S + (1-cos(phi)) * S @ S
    return R

def rot_to_ax_phi(R):
    if abs(np.trace(R)-3) < 1e-10:
        return np.array([0, 0, 0]), 2*pi

    if abs(np.trace(R)+1) < 1e-10:
        phi = pi
    else:
        phi = acos((np.trace(R)-1)/2)

    eigv, v = eig(R)
    for i in range(3):
        if is_one(eigv[i]):
            index0 = i
    index1 = (i+1)%3
    index2 = (i+2)%3
    s = [v[0, index0].real, v[1, index0].real, v[2, index0].real]
    return s, phi

def is_one(number):
    if abs(number-1) < 1e-10:
        return True

s = get_unit_vector()
phi = pi/3
R = get_rotation_matrix(s, phi)

print(s)
print(phi)
rot_to_ax_phi(R)
```

```
[ 0.16573982 -0.58044732  0.79725229]
1.0471975511965976
```

Out[154]:
```
([0.16573981989433331, -0.5804473225914154, 0.79725229243812856],
 1.0471975511965974)
```

In [ ]: