Algorithm to compute orthonormality condition of Legendre polynomial $P_l(x)$

1. Read m and n

2. if $m = n$ then

$$\int_{-1}^{1} P_m(x)P_n(x)dx = \frac{2}{2n+1}$$

3. if $m \neq n$ then

$$\int_{-1}^{1} P_m(x)P_n(x)dx = 0$$

```
"""
Program to show orthonormality relation of Legendre Polynomials.
Use of quad function from scipy.
Quadrature means the process of determining area.
@author: User
"""

from scipy.special import legendre
from scipy.integrate import quad

def integrand(x,m,n):
    Pm = legendre(m)
    Pn = legendre(n)
    Fm = Pm(x)
    Fn = Pn(x)
    return Fm*Fn

m = 2
n = 2
I = quad(integrand, -1, 1, args=(m,n))
#I is tuple -> I[0] = value of Integral, I[1] = Error

# From orthonormality condition when n != m --> Integral = 0
# When n == m --> Integral = 2/(2n+1)
if m == n:
    exactValue = 2.0/(2.0*n+1)
else:
    exactValue = 0
# From orthonormality condition when n != m --> Integral = 0
# When n == m --> Integral = 2/(2n+1)
print('m =', m, 'n = ', n)
print('Integral of Pm*Pn = ', I[0])
print('Orthonormal value of integral of Pm*Pn = ', exactValue)
```

```
m = 2 n =  2
Integral of Pm*Pn =  0.4
Orthonormal value of integral of Pm*Pn =  0.4
```