Codes du TP8

```
import numpy as np

#Un premier exemple

def approximation(eps):
    n=1
    u=1
    v=2
    while np.abs(u-v)>eps:
        n=n+1
        u=u+1/n**2
        v=u+1/n
    return u,v,n

#Exercice 2.1

#Question 2
def exo1(n):
    u=2/3
    v=1
    for k in range(2,n+1):
        u=u+2/((4*(k-1)+1)*(4*(k-1)+3))
        v=u+1/(4*k-1)
    return u,v

#Question 3, méthode 1
def approximation1(eps):
    k=1
    u=2/3
    v=1
    while np.abs(u-v)>eps:
        k=k+1
        u=u+2/((4*(k-1)+1)*(4*(k-1)+3))
        v=u+1/(4*k-1)
    return u,v,k

#Question 3, méthode 2
def approximation1bis(eps):
    k=1
    u=2/3
    v=1
    while np.abs(u-v)>eps:
        u=u+2/((4*k+1)*(4*k+3))
        v=u+1/(4*(k+1)-1)
        k=k+1
    return u,v,k
```

```
#Exercice 2.2

def exo2(n):
    a=1
    b=2
    for k in range(1,n+1):
        anew=np.sqrt(a*b)
        bnew=(a+b)/2
        a=anew
        b=bnew
    return a,b

def approximation2(eps):
    n=0
    a=1
    b=2
    while np.abs(a-b)>eps:
        n=n+1
        anew=np.sqrt(a*b)
        bnew=(a+b)/2
        a=anew
        b=bnew
    return a,b,n

#Exercice 2.3

def exo3(n):
    v=1
    w=2
    for k in range(1,n+1):
        vnew=(2/3)*v+(1/3)*w
        wnew=(1/3)*v+(2/3)*w
        v=vnew
        w=wnew
    return v,w

def approximation3(eps):
    n=0
    v=1
    w=2
    while np.abs(w-v)>eps:
        n=n+1
        vnew=(2/3)*v+(1/3)*w
        wnew=(1/3)*v+(2/3)*w
        v=vnew
        w=wnew
    return v,w,n
```