

## Correction de la série des fonctions

### Exo 1:

```
def signe(x):
    if x > 0 :
        return 1
    elif x < 0 :
        return -1
    return 0
```

```
def valAbs(x):
    return x*signe(x)
```

```
def pgcd(a,b):
    while b!=0:
        a,b = b, a%b
    return valAbs(a)
```

```
def ppcm(a,b):
    return valAbs(a*b)/pgcd(a,b)
```

```
# Pogramme principal
print("Donner x :")
x=int(input())
print("Donner y :")
y=int(input())
print("Le pgcd est : ", pgcd(x,y))
print("Le ppcm est : ",ppcm(x,y))
```

### Exo 2:

```
def f(x):
    return 2*x**3 +x - 5

def tabuler(borneInf, borneSup, pas):
    i=borneInf
    while i<=borneSup :
        print(f(i))
        i=i+pas
```

### Exo 3:

```
def somme_div(n):
    s=0
    for i in range(1,n+1):
        if n%i==0:
            s=s+i
    return s

def parfait(m):
    if somme_div(m)==2*m :
        return True
    return False

def premier(x):
    return somme_div(x)==1+x:
```

```

def amis(x,y):
    if somme_div(x)-x==y and somme_div(y)-
y==x:
        return True
    return False

```

```

# Programm principal
print("Donner a :")
a=int(input())
print("Donner b :")
b=int(input())

```

```

if parfait(a) :
    print("a est parfait")
if parfait(b) :
    print("b est parfait")
if amis(a,b)==True :
    print("a et b sont amis")
else :
    print("a et b ne son pas amis")

```

#### **Exo 4 :**

```

def fact(n) :
    f=1
    for i in range(2,n+2):
        f=f*i
    return f

```

```

def comb(n,p):
    return fact(n)/(fact(p)*fact(n-p))

```

```

def pascal(nl):
    for i in range(0,nl):
        for j in range(0,i+1):
            print(comb(i,j), "\t", end="")
        print()

```

#### **Exo5 :**

```

def nb_chiffres(n):
    if n== 0: return 1
    s=0
    while n!=0 :
        n=n//10
        s=s+1
    return s

```

```

def armstrong(n):
    a = n
    nb = nb_chiffres(n)
    s=0
    while a!=0 :
        u = a % 10
        s = s + u**nb
        a = a // 10
    return s==n

```

### **Exo 6:**

```
def conversionHMS(heure, minute, seconde) :
    return seconde + minute*60 + heure*3600

def extraire(chaine, numeroDebut, numeroFin) :
    return chaine[numeroDebut-1, numeroFin]

def conversionChiane(horaire) :
    h = int(extraire(horaire, 1,2))
    m = int(extraire(horaire, 4,5))
    s = int(extraire(horaire, 7,8))
    return conversionHMS(h,m,s)

def radar(H0 , H1, distance, limite) :
    min_time = distance / limite
    min_time_sec = min_time * 3600
    duree = conversionChiane(H1) –
conversionChiane(H0)
    return duree < min_time_sec
```

### **Exo 7:**

```
def de( ) :
    v = int(input("Choisir un nombre entre 1 et 6 : "))
    while v<1 or v>6 :
        v = int(input("Veuillez réessayer : "))
    return v
```

```
a1 = de( )
a2 = de( )
print("Premier lancé : ", a1, "Deuxième lancé : ", a2)

print("Début du test : ")
N = 1000
n1=0 # nombre de 1 obtenu
for i in range(1 , N+1) :
    print("Lancé numéro ", i , " :")
    n = de()
    if n==1 :
        n1 = n1 + 1
p1 = n1/N

if abs(p1-1/6) < 0.05*(1/6) :
    print("Le dé n'est pas pipé")
else :
    print("Le dé est pipé")
```