

Informatique en CPGE (2018-2019)
Corrigé exercices 4

Exercice 1

```
def contient_e(chaine):
    if 'e' in chaine:
        return True
    else:
        return False

# ou bien
def contient_e(chaine):
    for c in chaine:
        if c=='e':
            return True
    return False
```

Exercice 2

```
def compteur_e(chaine):
    cpt=0
    for c in chaine:
        if c=='e':
            cpt+=1
    return cpt
```

Exercice 3

```
def taret(ch):
    chtaret=ch[0]
    for i in range(1,len(ch)):
        chtaret+='-'+ch[i]
    return chtaret
```

Exercice 4

```
def inverse(ch):
    if ch==[]:
        return []
    chinverse=[]
    for i in range(len(ch)):
        chinverse+=ch[-1-i]
    return chinverse

# ou bien
def inverse(ch):
    return ch[-1::-1] if len(c)!=0 else []
```

Exercice 5

```
# question 1
def somme_diviseurs(n):
    s=0
    for d in range(1,n+1):
        if n%d==0:
            s=s+d
    return s

# question 2
def parfait(n):
    s=somme_diviseurs(n)-n
    return s==n
```

Exercice 6

```
# question 1
def suite(n):
    u=0
    for k in range(n):
        u=3*u-2*k+3
    return u
# pour N= 3, u=29

# question 2
def seuil(p):
    u=0
    n=0
    while u<10**p:
        u=3*u-2*n+3
        n+=1
    return n
```

Exercice 7

```
# question 1
def somme(n):
    u=0
    for i in range(1,n+1):
        u=u+1/i
    return u

# question 2
from math import log
def suite(n):
    u=0
    for i in range(1,n+1):
        u=u+1/i
    return u-log(n)
```

Exercice 8

```
# loi binomiale B(n,p) avec n=9 et p=2/7
from random import randint

def test():
    c=0
    for i in range(9):
        a=randint(1,7)
        if a>5:
            c+=1
    return c
```

Exercice 9

```
def seuil(k):
    u, j=0,1
    while u<0.05-10**(-k):
        u=0.2*u+0.04
        j+=1
    return j
# renvoie le rang à partir duquel les termes de la suite
# sont proches de la limite 0.05, à moins de 10**(-k)
```