

TP 01

Découverte de la programmation fonctions

Savoirs et compétences :

- Th. 0 : Connaître les bases de l'algorithmique et de la programmation (variable, types, structures, fonctions).

Proposition de corrigé

Activité 1 – Environnement de développement intégré Python et prise en main élémentaire de Python**Activité 2 – Types composés****Activité 3 – Variables****Activité 4 – Structures de boucles****Exercice 1 –****Question 1**

```
def nb_exos():  
    """Nombre d'exercices réalisés en 240 min"""  
    t=0  
    i=0  
    #0 exos réalisés en 0 minute  
    while t<=240:  
        #invariant de boucle i exos réalisés en t<=240 minutes  
        i=i+1  
        #invariant de boucle (i-1) exos réalisés en t<=240 minutes  
        t=t+(i)**(1/2)  
    # invariant de fin de boucle : i+1 exos réalisé en t minutes  
    #sortie de boucle : n<=240<t avec n le nombre de minutes pour faire i exos-1  
    return i-1
```

Activité 5 – Structures de boucles**Activité 6 – Simulation d'un prêt immobilier****Exercice 2 –****Question 1**

```
def reste_a_payer(p,t,m,d):  
    """p = montant du pret en euros  
    t = taux mensuel  
    m = mensualites  
    d = duree en annees
```

```

    Calcule le montant restant a payer a l'echeance du pret"""
    dette = p
    for mois in range(d*12):
        # Inv : dette est d'au début du mois
        dette = dette*(1+t)-m
    return dette

```

Question 2

```

def somme_totale_payee(p,t,m,d):
    """p = montant du pret
       t = taux
       m = mensualites
       d = duree en annees
       Calcule le montant total paye"""
    return reste_a_payer(p,t,m,d) + 12*d*m

```

Question 3

```

def cout_total(p,t,m,d):
    """p = montant du pret
       t = taux
       m = mensualites
       d = duree en annees
       Calcule le cout total du credit"""
    return somme_totale_payee(p,t,m,d) - p

```

Question 4

```

def duree_mensualite(p,t,m):
    """Durée du prêt
       p = montant prêté
       t = taux mensuel
       m = mensualité"""
    emprunt = p
    d = 0
    while (1+t)*emprunt >= m:
        # Inv : emprunt est d'au début du mois d
        d = d+1
        emprunt = (1+t)*emprunt-m
    return d

```

Question 5

Si la mensualité est trop petite la dette augmentera plus vite que le capital restant du diminuera et ainsi la condition de la boucle conditionnelle ne sera jamais vérifiée et la boucle tournera à l'infini.

Question 6

```

import matplotlib.pyplot as plt
def tracer_mensualite(p,t,m):
    """Trace
       p = montant prêté
       t = taux mensuel
       m = mensualité"""
    emprunt = p
    d = 0
    mois=[]#numero de mensualite
    capital_rd=[]#capital restant du
    interets=[]
    while (1+t)*emprunt >= m:
        # Inv : emprunt est d'au début du mois d
        d = d+1
        emprunt = (1+t)*emprunt-m

```

```
mois.append(d)
capital_rd.append(emprunt)
interets.append(t*emprunt)
plt.clf()
plt.plot(mois, capital_rd)
plt.xlabel('Mensualité')
plt.ylabel('capital restant du')
plt.savefig('capital_restant_du.png')
plt.clf()
plt.plot(mois, interets, 'b')
plt.xlabel('Mensualité')
plt.ylabel('Intérêts')
plt.savefig('interets.png')
```

