Vous soignerez la présentation et encadrerez vos résultats. Les copies ne respectant pas ces consignes auront 0,5 point en moins !

Exercice I **(*9,5 points*)**

Dans une grande ville française, des trottinettes électriques sont mises à disposition des usagers. Une entreprise, chargée de l'entretien du parc de trottinettes, contrôle leur état chaque lundi.

On estime que :

— lorsqu'une trottinette est en bon état un lundi, la probabilité qu'elle soit encore en bon état le lundi suivant est $0,9$;

— lorsqu'une trottinette est en mauvais état un lundi, la probabilité qu'elle soit en bon état le lundi suivant est $0,4$.

On s'intéresse à l'état d'une trottinette lors des phases de contrôle.
Soit n un entier naturel. On note B_n l'évènement « la trottinette est en bon état n semaines après sa mise en service »

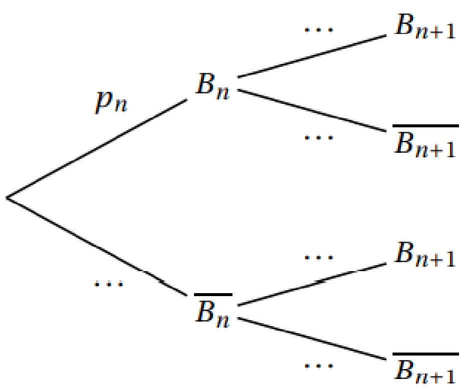
Enfin, on note u_n la probabilité de l'événement B_n, c'est-à-dire : $u_n = p(B_n)$.

Lors de sa mise en service, la trotinnette est en bon état : on a donc $u_0 = 1$.

1) *a)* Combien vaut u_1 ?

b) A l'aide d'un arbre de probabilité, montrer que $u_2 = 0,85$.

2. Recopier et compléter l'arbre pondéré ci-dessous :



3. Déduire de l'arbre que pour tout entier naturel n, $u_{n+1} = 0,5u_n + 0,4$.

4. On définit, pour tout entier naturel n, la suite (V_n) par : $V_n = u_n - 0,8$.

a) Démontrer que (V_n) est une suite géométrique de raison 0,5. Préciser son premier terme.

b) Exprimer V_n en fonction de n, puis u_n en fonction de n.

c) A l'aide de votre calculatrice, déterminer la limite de la suite (u_n), et interpréter cette dernière dans le contexte de l'exercice.

d) Un publicitaire annonce que, quelle que soit la semaine, la trottinette est en bon état plus de trois fois sur quatre. A-t-il raison ?

5) On considère l'algorithme suivant :

```
def etattrotinnette():
    u=1
    n=0
    while u > 0.85 :
        u=0.5*u+0.4
        n=n+1
    return (n)
```

a) Quel est le rôle de ce programme ? Interprétez dans le cadre de l'exercice.

b) Déterminer, à la main, la valeur affichée en sortie par cet algorithme.

<u>Exercice II</u> (9,5 points)

Un apiculteur étudie l'évolution de sa population d'abeilles. Au début de son étude, il évalue à 10 000 le nombre de ses abeilles.
Chaque année, l'apiculteur observe qu'il perd 20 % des abeilles de l'année précédente.

Pour compenser cette perte, il remet ***<u>chaque année</u>*** 10000 nouvelles abeilles dans la ruche.

On note u_0 le nombre d'abeilles, en dizaines de milliers, de cet apiculteur au début de l'étude.
Pour tout entier naturel n non nul, u_n désigne le nombre d'abeilles, en dizaines de milliers, au bout de la n-ième année.

1) Que vaut u_0 ?

2) Expliquer pourquoi, pour tout entier naturel n, on a : $u_{n+1} = 0{,}8\, u_n + 1$.

3) ***<u>A l'aide de votre calculatrice</u>***, complétez le tableau suivant :

n	0	1	2	3	10	50
u_n						

i) Quelle conjecture émettez-vous concernant le sens de variation de la suite (u_n) ?

ii) Quelle conjecture émettez-vous sur la limite éventuelle de la suite (u_n) ?

4a) Soit (V_n) la suite définie, pour tout entier naturel n, par : $V_n = u_n - 5$.

Démontrer que (V_n) est une suite géométrique, dont on précisera la raison et le premier terme.

4b) En déduire avec soin que pour tout entier naturel n, $u_n = 5 - 4 \times 0.8^n$.

4c) Etudier alors avec soin le sens de variation de la suite (u_n).

4d) Trouver à l'aide de votre calculatrice la limite de la suite (u_n). Interprétez ce résultat dans le contexte de l'exercice.

5) Compléter l'algorithme ci-dessous afin qu'il renvoie en sortie le plus petit entier n à partir duquel $u_n > 3$.

```python
def grosseruche():
    n=.....
    while ................. :
        n=......
    return(...)
```

Donner la valeur de ce plus petit entier *n* obtenu après codage de l'algorithme, et interpréter cette valeur dans le cadre de l'exercice.

***<u>Exercice III</u>* (1 point)**

Aucune justification n'est demandée dans cette question de type QCM. Une seule réponse est exacte.

On pose $S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \ldots + \frac{1}{100}$.

Parmi les scripts Python ci-dessous, celui qui permet de calculer la somme S est :

a.

```
def somme_a() :
    S = 0
    for k in range(100) :
        S=1/(k+1)
    return S
```

b.

```
def somme_b() :
    S = 0
    for k in range(100) :
        S = S + 1/(k + 1)
    return S
```

c.

```
def somme_c() :
    k = 0
    while S < 100 :
        S = S+1/(k+1)
    return S
```

d.

```
def somme_d() :
    k = 0
    while k < 100 :
        S = S + 1/(k + 1)
    return S
```