Une puce sur un piano infini

Soit $p \in]0;1[$.
Une puce saute sur les touches d'un piano infini (dans les aigus). Les touches du piano sont numérotées du grave vers l'aigu en partant de 0. La puce commence sur la première touche puis se déplace vers les aigus par bonds successifs. A chaque saut, elle se déplace soit d'une touche, avec probabilité p, soit de deux touches, avec probabilité $1-p$. On suppose que les sauts sont indépendants.

Pour tout $n \in \mathbb{N}$, on considère les variables aléatoires suivantes :

- S_n désigne le nombre de fois où la puce choisit de sauter d'une touche au cours de ses n premiers bonds ;
- X_n désigne le nombre de touches parcourues après n bonds ;
- Y_n désigne le nombre de bonds nécessaires à la puce pour avancer de n touches.

Pour les questions `Python`, on suppose que la bibliothèque numpy.random a été importée au préalable via la commande : `import numpy.random as rd`.

Soit n un entier naturel.

1. Ecrire une fonction Python `bond(p)` qui simule un bond de la puce et renvoie 1 si la puce se déplace d'une touche, 2 si elle se déplace de deux touches.
2. En utilisant la fonction `bond(p)`, écrire une fonction `touches_touchées(n,p)` qui renvoie une liste contenant toutes les touches que la puce a touchées après n sauts.
3. Ecrire une fonction `Python` qui simule la variable aléatoire X_n. On pourra s'aider d'une fonction précédente.
4. Déterminer la loi de la variable aléatoire S_n.
5. Ecrire une fonction `Python` qui simule la variable aléatoire S_n. Cette fonction prendra en entrée deux paramètres. (*On pourra créer une petite fonction auxiliaire si besoin, que l'on pourra utiliser dans la fonction qui simule S_n, mais on peut très bien faire sans.*)

6. (a) Exprimer X_n en fonction de S_n.
   (b) En déduire l'espérance et la variance de la variable aléatoire X_n.
   (c) Ecrire une fonction `Python` qui estime l'espérance de la variable X_n. On pourra par exemple d'abord construire une liste qui contient un grand nombre de simulations de X_n. Comment optimiser cette estimation ?
7. Quel est l'image de la variable aléatoire Y_n ?
   On vérifiera en particulier que $Y_n(\Omega) \subseteq [\![0;n]\!]$.
8. (a) Pour tout $k \in [\![0;n+1]\!]$, exprimer $\mathbb{P}(Y_{n+2} = k+1)$ en fonction de $\mathbb{P}(Y_{n+1} = k)$ et de $\mathbb{P}(Y_n = k)$.
   On pourra appliquer la formule des probabilités totales au système complet d'événements $\{(X_1 = 1);(X_1 = 2)\}$.
   (b) En déduire la relation de récurrence suivante :

$$\forall n \in \mathbb{N}, \mathbb{E}(Y_{n+2}) = p\mathbb{E}(Y_{n+1}) + (1-p)\mathbb{E}(Y_n) + 1.$$

9. (a) Trouver un réel c tel que la suite $(u_n)_{n\in\mathbb{N}}$ définie par

$$\forall n \in \mathbb{N}, u_n = \mathbb{E}(Y_n) - cn$$

soit récurrente linéaire d'ordre deux.

(b) En déduire, pour tout $n \in \mathbb{N}$, l'espérance de la variable aléatoire Y_n.

(c) Donner un équivalent de la suite $(\mathbb{E}(Y_n))_{n\in\mathbb{N}}$.

Corrigé

1. ```
   import numpy.random as rd
   def bond(p):
       u = rd.random()
       if u < p:
          return 1
       else:
          return 2
   ```
2. ```
   def touches_touchées(n,p):
       l=[0]
       position=0
       for k in range (n):
           position+=bond(p)
           l.append(position)
       return l
   ```
3. ```
   def X_n(n,p): #nb de touches PARCOURUES (pas touchéees forcément)
       s=0
       for k in range(n):
           s+=bond(p)
       return s
   #ou marche aussi: #derniere touche touchée !
   def X_n2(n,p):
       return touches_touchées(n,p)[-1]
   ```
4. On reconnaît une loi binomiale : en effet, si on considère chaque saut comme une expérience de Bernoulli dont le succès (représenté par le fait que la puce saute d'une touche) est réalisé avec une probabilité p, et puisque les sauts sont indépendants, on en déduit que S_n suit une loi binomiale de paramètres (n, p), i.e. $\boxed{S_n \hookrightarrow \mathcal{B}(n, p).}$
5. ```
   def S_n(n,p):
       s=0
       for k in range(n):
           r=rd.random()
           if r<p:
               s+=1
       return s

   #avec une fonction auxiliaire
   def un_saut(p):#renvoie 1 si saut de 1 touche, 0 si saut de 2
       r=rd.random()
       if r<p:
           return 1
       else:
           return 0



   def S_n2(n,p):
       return sum([un_saut(p) for k in range(n)])

   #ou encore, sans sum:
   ```

```python
def S_n3(n,p):
    l=[un_saut(p) for k in range(n)]
    succes=0
    for element in l: #on balaye sur les éléments et pas sur les indices
        succes+=element
    return succes

#ou encore
def S_n4(n,p):
    l=[un_saut(p) for k in range(n)]
    succes=0
    for k in range(len(l)): #on balaye sur les indices
        succes+=l[k]
    return succes
```

6. (a) Après n bonds, la puce aura réalisé S_n sauts d'une touche et $(n - S_n)$ sauts de 2 touches donc le nombre de touches parcourues après n sauts vaut

$$\boxed{X_n = S_n + 2(n - S_n) = 2n - S_n.}$$

(b) • Par linéarité de l'espérance, on a $\mathbb{E}(X_n) = 2n - \mathbb{E}(S_n)$.
Puisque $S_n \hookrightarrow \mathcal{B}(n,p)$, on sait que $\mathbb{E}(S_n) = np$ donc

$$\boxed{\mathbb{E}(X_n) = 2n - np = n(2 - p).}$$

• Par propriété de la variance, puisque $2n$ est une constante, on a

$$V(X_n) = V(-S_n + 2n) = (-1)^2 V(S_n) = V(S_n).$$

Puisque $S_n \hookrightarrow \mathcal{B}(n,p)$, on sait que $V(S_n) = np(1-p)$ donc

$$\boxed{V(X_n) = np(1-p).}$$

(c)
```python
def esp_Xn(n,p):
    real_X=[X_n(n,p) for k in range (1000)]
    return sum(real_X)/len(real_X)
```

7. Au maximum, si la puce choisit de ne sauter qu'une touche à la fois, elle devra faire n bonds pour avancer de n touches donc $Y_n \leqslant n$.
Au minimum, c'est à dire si la puce décide de sauter de deux touches à chaque fois, elle devra faire $\frac{n}{2}$ bonds pour avancer de n touches si n est pair ; si n est impair, $n-1$ est pair et elle pourra parcourir les $n-1$ premières touches en $\frac{n-1}{2}$ bonds et il lui restera un saut d'une touche à effectuer, ce qui donne en tout $\frac{n-1}{2} = \frac{n+1}{2}$ bonds.
Enfin, la variable aléatoire Y_n peut prendre comme valeur tous les entiers entre sa valeur minimale et sa maximale (puisque la puce peut faire des sauts d'une seule touche).
Donc $Y_n(\Omega) = [\![\frac{n}{2}, n]\!]$ si n est pair et $[\![\frac{n+1}{2}, n]\!]$ si n est impair, ce qu'on peut résumer en

$$\boxed{Y_n(\Omega) = \left[\!\!\left[\left\lfloor \frac{n+1}{2} \right\rfloor, n \right]\!\!\right] \subset [\![0, n]\!].}$$

8. (a) Soit $k \in [\![0, n+1]\!]$.

Notons que les événements $(X_1 = 1)$ et $(X_1 = 2)$ forment bien un système complet d'événements puisque le nombre de touches parcourues après un seul bond est nécessairement égal à 1 ou 2.
D'après la formule des probabilités totales dans ce système complet d'événements, on a

$$\mathbb{P}(Y_{n+2} = k+1) = \mathbb{P}(X_1 = 1)\mathbb{P}_{(X_1=1)}\mathbb{P}(Y_{n+2} = k+1) + \mathbb{P}(X_1 = 2)\mathbb{P}_{(X_1=2)}\mathbb{P}(Y_{n+2} = k+1).$$

La variable aléatoire X_1 désigne le nombre de touches parcourues au premier bond, donc d'après l'énoncé, $\mathbb{P}(X_1 = 1) = p$ et $\mathbb{P}(X_1 = 2) = 1 - p$.
Par ailleurs, $\mathbb{P}_{(X_1=1)}\mathbb{P}(Y_{n+2} = k + 1)$ désigne la probabilité que la puce ait eu besoin de $k + 1$ bonds pour parcourir $n + 2$ touches sachant qu'elle en avait parcouru une après le premier bond. Il lui restait donc $n + 1$ touches à parcourir lors des k bonds suivants, et puisque les sauts sont indépendants, on en déduit que la probabilité de parcourir $n + 1$ touches en k sauts vaut $\mathbb{P}(Y_{n+1} = k)$.
Ainsi, on a $\mathbb{P}_{(X_1=1)}\mathbb{P}(Y_{n+2} = k + 1) = \mathbb{P}(Y_{n+1} = k)$.
De même, $\mathbb{P}_{(X_1=2)}\mathbb{P}(Y_{n+2} = k + 1)$ désigne la probabilité que la puce ait eu besoin de $k + 1$ bonds pour parcourir $n + 2$ touches sachant qu'elle en avait parcouru deux après le premier bond. Il lui restait donc n touches à parcourir lors des k bonds suivants, et puisque les sauts sont indépendants, on en déduit que la probabilité de parcourir n touches en k sauts vaut $\mathbb{P}(Y_n = k)$.
On obtient donc

$$\boxed{\mathbb{P}(Y_{n+2} = k + 1) = p\mathbb{P}(Y_{n+1} = k) + (1 - p)\mathbb{P}(Y_n = k).}$$

(b) D'après la question 3, $Y_{n+2}(\Omega) \subset [\![0, n + 2]\!]$ donc

$$\begin{aligned}
\mathbb{E}(Y_{n+2}) &= \sum_{i=0}^{n+2} i\mathbb{P}(Y_{n+2} = i) \\
&= \sum_{i=1}^{n+2} i\mathbb{P}(Y_{n+2} = i) \quad \text{(car le terme pour } i = 0 \text{ est nul)} \\
&= \sum_{k=0}^{n+1} (k + 1)\mathbb{P}(Y_{n+2} = k + 1) \quad \text{(en posant le changement d'indice } k = i - 1) \\
&= \sum_{k=0}^{n+1} k\mathbb{P}(Y_{n+2} = k + 1) + \sum_{k=0}^{n+1} \mathbb{P}(Y_{n+2} = k + 1) \quad \text{(par linéarité de la somme).}
\end{aligned}$$

• Calculons $\displaystyle\sum_{k=0}^{n+1} \mathbb{P}(Y_{n+2} = k + 1)$.

En posant $j = k + 1$, on obtient $\displaystyle\sum_{k=0}^{n+1} \mathbb{P}(Y_{n+2} = k + 1) = \sum_{j=1}^{n+2} \mathbb{P}(Y_{n+2} = j)$.
Or, $\mathbb{P}(Y_{n+2} = 0) = 0$, puisque si $n \in \mathbb{N}$, le nombre de bonds nécessaires pour avancer de $n + 2$ touches (avec $n \geqslant 2$) ne peut pas valoir 0, donc $\displaystyle\sum_{j=1}^{n+2} \mathbb{P}(Y_{n+2} = j) = \sum_{j=0}^{n+2} \mathbb{P}(Y_{n+2} = j)$ puisqu'on a rajouté un terme nul.

Or, $Y_{n+2}(\Omega) \subset [\![0, n+2]\!]$ donc $\displaystyle\sum_{j=0}^{n+2} \mathbb{P}(Y_{n+2} = j) = 1$, d'où $\displaystyle\sum_{k=0}^{n+1} \mathbb{P}(Y_{n+2} = k+1) = 1$.

• Calculons $\displaystyle\sum_{k=0}^{n+1} k\mathbb{P}(Y_{n+2} = k+1)$.

En utilisant la question précédente, on obtient

$$\begin{aligned}
\sum_{k=0}^{n+1} k\mathbb{P}(Y_{n+2} = k+1) &= \sum_{k=0}^{n+1} k(p\mathbb{P}(Y_{n+1} = k) + (1-p)\mathbb{P}(Y_n = k)) \\
&= p\sum_{k=0}^{n+1} k\mathbb{P}(Y_{n+1} = k) + (1-p)\sum_{k=0}^{n+1} k\mathbb{P}(Y_n = k) \\
&= p\mathbb{E}(Y_{n+1}) + (1-p)\sum_{k=0}^{n} k\mathbb{P}(Y_n = k) \quad (\text{car}\, Y_n \subset [\![0, n]\!]) \\
&= p\mathbb{E}(Y_{n+1}) + (1-p)\mathbb{E}(Y_n).
\end{aligned}$$

Finalement, on obtient

$$\boxed{\forall n \in \mathbb{N}, \mathbb{E}(Y_{n+2}) = p\mathbb{E}(Y_{n+1}) + (1-p)\mathbb{E}(Y_n) + 1.}$$

9. (a) Soit $c \in \mathbb{R}$. On pose pour tout $n \in \mathbb{N}, u_n = \mathbb{E}(Y_n) - cn$.
D'après la question précédente, on a pour tout $n \in \mathbb{N}$,

$$\begin{aligned}
u_{n+2} &= \mathbb{E}(Y_{n+2}) - c(n+2) \\
&= p\mathbb{E}(Y_{n+1}) + (1-p)\mathbb{E}(Y_n) + 1 - cn - 2c \\
&= p(u_{n+1} + c(n+1)) + (1-p)(u_n + cn) + 1 - cn - 2c \\
&= pu_{n+1} + (1-p)u_n + pcn + pc + cn - pcn + 1 - cn - 2c \\
&= pu_{n+1} + (1-p)u_n + 1 + pc - 2c.
\end{aligned}$$

Pour que la suite $(u_n)_{n\in\mathbb{N}}$ soit récurrente linéaire d'ordre deux, il faut et il suffit que $1 + pc - 2c = 0$, i.e. $\boxed{c = \dfrac{1}{2-p}}$ (ce qui est possible car $2 - p > 1$).

(b) • D'après la question précédente, pour $c = \dfrac{1}{2-p}$, la suite $(u_n)_{n\in\mathbb{N}}$ vérifie la relation de récurrence linéaire d'ordre deux suivante :

$$\forall n \in \mathbb{N}, u_{n+2} - pu_{n+1} + (p-1)u_n = 0.$$

On considère $(E) : r^2 - pr + (p-1)$ l'équation caractéristique associée dont les racines sont $r_1 = 1$ et $r_2 = p - 1$ (qui sont bien distinctes car $p - 1 \neq 1$ puisque $p \neq 2$). Ainsi, il existe $(\lambda, \mu) \in \mathbb{R}^2$ tel que pour tout $n \in \mathbb{N}$,

$$u_n = \lambda \times 1^n + \mu(p-1)^n = \lambda + \mu(p-1)^n,$$

d'où pour tout $n \in \mathbb{N}$,

$$\mathbb{E}(Y_n) = u_n + cn = \lambda + \mu(p-1)^n + \frac{n}{2-p}.$$

On détermine λ et μ grâce à $\mathbb{E}(Y_0)$ et $\mathbb{E}(Y_1)$.

D'après l'énoncé, il est clair que $Y_0 = 0$ et $Y_1 = 1$ donc $\mathbb{E}(Y_0) = 0$ et $\mathbb{E}(Y_1) = 1$, et on obtient

$$\left\{\begin{array}{ccc} \lambda+\mu & = & 0 \\ \lambda+\mu(p-1)+\dfrac{1}{2-p} & = & 1 \end{array}\right. \Leftrightarrow \left\{\begin{array}{ccc} \mu & = & -\lambda \\ \lambda(2-p) & = & \dfrac{1-p}{2-p} \end{array}\right. \Leftrightarrow \left\{\begin{array}{ccc} \lambda & = & \dfrac{1-p}{(2-p)^2} \\ \mu & = & \dfrac{p-1}{(2-p)^2} \end{array}\right.$$

donc

$$\boxed{\forall n \in \mathbb{N}, \mathbb{E}(Y_n) = \frac{1-p}{(2-p)^2} + \frac{(p-1)^{n+1}}{(2-p)^2} + \frac{n}{2-p}.}$$

(c) Puisque $p \in]0,1[, 2-p > 0$ donc $\lim\limits_{n\to+\infty} \dfrac{n}{2-p} = +\infty$.

Par ailleurs, $p-1 \in]-1,0[$ donc $\lim\limits_{n\to+\infty} (p-1)^{n+1} = 0$.

Ainsi, pour tout $n \in \mathbb{N}^*$, on a

$$\mathbb{E}(Y_n) = \frac{n}{2-p}\left(1 + \frac{1-p}{n(2-p)} + \frac{(p-1)^{n+1}}{n(2-p)}\right)$$

avec $\lim\limits_{n\to+\infty} \dfrac{1-p}{n(2-p)} = \lim\limits_{n\to+\infty} \dfrac{(p-1)^{n+1}}{n(2-p)} = 0$ donc

$$\lim_{n\to+\infty} 1 + \frac{1-p}{n(2-p)} + \frac{(p-1)^{n+1}}{n(2-p)} = 1.$$

On en conclut que $\boxed{\mathbb{E}(Y_n) \underset{+\infty}{\sim} \dfrac{n}{2-p}.}$