Corrigé du DS n° 4 de mathématiques et d'informatique

Problème

Soit $n \in \mathbb{N}^*$. Ce problème propose de dénombrer l'ensemble $\mathscr{E}_n$ des parties de $[\![1,n]\!] = \{1,2,\dots,n\}$ ne contenant pas d'entiers consécutifs. Par exemple pour $n=3$, l'ensemble des parties de $[\![1,3]\!]$ est :

$$\mathscr{P}\big([\![1,3]\!]\big) = \Big\{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\Big\}$$

ce qui donne $\operatorname{card}(\mathscr{E}_3) = 5$ car les parties $\{1,2\}$, $\{2,3\}$ et $\{1,2,3\}$ contiennent des entiers consécutifs. Pour chaque $k \in \mathbb{N}$, on notera $\mathscr{E}_{n,k}$ le sous-ensemble de $\mathscr{E}_n$ des parties contenant exactement k entiers. Ainsi :

$$\mathscr{E}_{3,0} = \Big\{\emptyset\Big\}, \quad \mathscr{E}_{3,1} = \Big\{\{1\},\{2\},\{3\}\Big\}, \quad \mathscr{E}_{3,2} = \Big\{\{1,3\}\Big\} \quad \text{et} \quad \mathscr{E}_3 = \mathscr{E}_{3,0} \cup \mathscr{E}_{3,1} \cup \mathscr{E}_{3,2}.$$

Chaque partie de ce problème est indépendante des autres. Pour les questions d'informatique (parties B et D), on utilisera seulement le langage Python et on pourra faire appel aux fonctions des questions précédentes même si elles n'ont pas été écrites.

A) Préliminaires

1. *Que valent $\operatorname{card}(\mathscr{E}_1)$ et $\operatorname{card}(\mathscr{E}_2)$?*

   ► L'ensemble des parties de $[\![1,1]\!] = \{1\}$ est $\mathscr{P}(\{1\}) = \{\emptyset, \{1\}\}$ donc $\boxed{\operatorname{card}\mathscr{E}_1 = 2}$ car aucune des deux parties ne contient d'entiers consécutifs. De même, l'ensemble des parties de $[\![1,2]\!] = \{1,2\}$ est $\mathscr{P}(\{1,2\}) = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$ donc $\boxed{\operatorname{card}(\mathscr{E}_2) = 3}$ car $\{1,2\}$ contient des entiers consécutifs.

2. (a) *Déterminer $\mathscr{E}_{4,0}$, $\mathscr{E}_{4,1}$, $\mathscr{E}_{4,2}$, $\mathscr{E}_{4,3}$ et $\mathscr{E}_{4,4}$.*

   ► L'ensemble des parties de $[\![1,4]\!] = \{1,2,3,4\}$ est :

$$\mathscr{P}\big([\![1,4]\!]\big) = \left\{\begin{array}{c} \emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}, \{2,4\}, \{3,4\}, \\ \{1,2,3\}, \{1,2,4\}, \{1,3,4\}, \{2,3,4\}, \{1,2,3,4\} \end{array}\right\}.$$

   Pensez à utiliser le cardinal pour vérifier rapidement que vous n'avez pas oublié une partie. Puisque $\operatorname{card}(\mathscr{P}([\![1,n]\!])) = 2^{\operatorname{card}([\![1,n]\!])} = 2^n$, vous devez obtenir $2^4 = 16$ parties.

   On remarque que les parties $\{1,2\}$, $\{2,3\}$, $\{3,4\}$, $\{1,2,3\}$, $\{1,2,4\}$, $\{1,3,4\}$, $\{2,3,4\}$ et $\{1,2,3,4\}$ contiennent des entiers consécutifs. Par conséquent :

$$\boxed{\mathscr{E}_{4,0} = \Big\{\emptyset\Big\}}, \ \boxed{\mathscr{E}_{4,1} = \Big\{\{1\},\{2\},\{3\},\{4\}\Big\}}, \ \boxed{\mathscr{E}_{4,2} = \Big\{\{1,3\},\{1,4\},\{2,4\}\Big\}} \text{ et } \boxed{\mathscr{E}_{4,3} = \mathscr{E}_{4,4} = \{\}}.$$

   Attention de ne pas confondre $\{\emptyset\}$ (l'ensemble qui contient une seule partie égale à $\emptyset$) et $\{\} = \emptyset$ (l'ensemble qui ne contient aucune partie).

   (b) *En déduire $\operatorname{card}(\mathscr{E}_4)$.*

   ► En sommant le nombre de parties obtenues à la question précédente, on obtient $1+4+3+0+0 = 8$ donc $\boxed{\operatorname{card}(\mathscr{E}_4) = 8}$.

3. *Que valent $\operatorname{card}(\mathscr{E}_{n,0})$, $\operatorname{card}(\mathscr{E}_{n,1})$ et $\operatorname{card}(\mathscr{E}_{n,n})$ lorsque $n \geqslant 2$?*

   ► On a $\mathscr{E}_{n,0} = \{\emptyset\}$ car $\emptyset$ est la seule partie contenant zéro entier, donc $\boxed{\operatorname{card}(\mathscr{E}_{n,0}) = 1}$. De même, on a $\mathscr{E}_{n,1} = \{\{1\},\{2\},\dots,\{n\}\}$ donc $\boxed{\operatorname{card}(\mathscr{E}_{n,1}) = n}$. La seule partie de $[\![1,n]\!]$ contenant n entiers est $\{1,2,\dots,n\}$ qui contient des entiers consécutifs lorsque $n \geqslant 2$, donc $\boxed{\operatorname{card}(\mathscr{E}_{n,n}) = 0}$.

4. *Déterminer* $\text{card}(\mathscr{E}_{n,2})$ *lorsque* $n \geqslant 3$. *Indication : distinguer plusieurs cas selon les valeurs possibles du plus petit entier de chaque partie appartenant à* $\mathscr{E}_{n,2}$.

▶ Les parties appartenant à $\mathscr{E}_{n,2}$ sont de la forme $\{x, y\}$ où x et y sont deux entiers différents et non consécutifs de $[\![1, n]\!]$. Si on pose x le plus petit de ces deux entiers, on a : $1 \leqslant x < y \leqslant n$ et $y \neq x + 1$, c'est-à-dire $x \in [\![1, n-2]\!]$ et $y \in [\![x+2, n]\!]$.
— Si $x = 1$, il y a $\text{card}([\![3, n]\!]) = n - 3 + 1 = n - 2$ choix possibles pour y.
— Si $x = 2$, il y a $\text{card}([\![4, n]\!]) = n - 4 + 1 = n - 3$ choix possibles pour y.
— Si $x = 3$, il y a $\text{card}([\![5, n]\!]) = n - 5 + 1 = n - 4$ choix possibles pour y.
— ...
— Si $x = n - 3$, il y a $\text{card}([\![n-1, n]\!]) = n - (n-1) + 1 = 2$ choix possibles pour y.
— Si $x = n - 2$, il y a $\text{card}([\![n, n]\!]) = n - n + 1 = 1$ seul choix possible pour y (qui est $y = n$).
En sommant tous ces choix possibles, on obtient :

$$\begin{aligned}\text{card}\left(\mathscr{E}_{n,2}\right) &= (n-2) + (n-3) + (n-4) + \cdots + 2 + 1 \\ &= \sum_{k=1}^{n-2} k = \boxed{\frac{(n-2)(n-3)}{2}} \quad \text{en reconnaissant la somme des premiers entiers.}\end{aligned}$$

B) Modélisation informatique

En Python, on modélisera chaque partie de $[\![1, n]\!]$ *par la liste de ses éléments. Par exemples,* $\{6, 3, 1, 5\}$ *sera modélisée par la liste* `[6,3,1,5]` *et* $\emptyset$ *par la liste vide* `[]`. *On rappelle que l'opérateur* `+` *permet de concaténer des listes. Par exemple,* `[2,7,1]+[4]` *renvoie* `[2,7,1,4]`.

5. (a) *Écrire une fonction* `chercher` *qui prend en arguments une liste d'entiers* `L` *et un indice* `i` *compris entre* `0` *et* `len(L)-1`, *puis qui renvoie le plus petit indice du plus petit élément de* `L` *dont l'indice est compris entre* `i` *et* `len(L)-1`. *Par exemples,* `chercher([3,6,1,5,2,7],3)` *renvoie* `4` *et* `chercher([1,2,1,4,1],2)` *renvoie* `2`.

▶ Par exemple :

```
def chercher(L,i):
    imin=i
    for j in range(i,len(L)):
        if L[j]<L[imin]:
            imin=j
    return imin
```

(b) *Écrire une fonction* `trier` *qui prend en argument une liste d'entiers* `L` *puis qui renvoie* `L` *après avoir trié ses éléments par ordre croissant. Par exemples,* `trier([5,3,2,6,3])` *renvoie* `[2,3,3,5,6]`. *Indication : commencer par échanger* `L[0]` *avec le plus petit élément de* `L`, *puis* `L[1]` *avec le plus petit élément dont l'indice est compris entre* `1` *et* `len(L)-1`, *puis* `L[2]` *avec le plus petit élément dont l'indice est compris entre* `2` *et* `len(L)-1`, *etc.*

▶ Par exemple :

```
def trier(L):
    for i in range(len(L)):
        imin=chercher(L,i)
        tmp=L[i]
        L[i]=L[imin]
        L[imin]=tmp
    return L
```

6. *Écrire une fonction* `tester` *qui prend en argument une liste d'entiers* `L`, *puis qui renvoie* `True` *si* `L` *contient des entiers deux à deux différents et non consécutifs, et* `False` *sinon. Par exemples,* `tester([8,2,5,2])` *et* `tester([4,2,1])` *renvoient* `False` *alors que* `tester([7,1,9,3])` *renvoie* `True`. *Indication : commencer par trier les éléments de* `L` *par ordre croissant.*

▶ Par exemple :

```
def tester(L):
    trier(L)
    for i in range(len(L)-1):
        if L[i+1]<L[i]+2:
            return False
    return True
```

On peut remplacer la condition «`if L[i+1]<L[i]+2`*» par «*`if L[i+1]<=L[i]+1`*», ou même par «*`if L[i+1]=L[i] or L[i+1]=L[i]+1`*». Le principe est de renvoyer* `False` *dès que deux entiers sont égaux, c'est-à-dire* `L[i+1]=L[i]`*, ou que deux entiers sont consécutifs, c'est-à-dire* `L[i+1]=L[i]+1`*.*

7. *On considère la fonction ci-dessous.*

```
def mystere(n):
    if n==1:
        return [[],[1]]
    else:
        Pold=mystere(n-1)
        Pnew=[]
        for i in range(len(Pold)):
            L=Pold[i]
            Pnew=Pnew+[L]
            L=L+[n]
            Pnew=Pnew+[L]
        return Pnew
```

(a) *Que renvoie* `mystere(1)` *?*

▶ Lorsque $n = 1$, la condition «`if n==1`» est vérifiée donc `mystere(1)` renvoie $\boxed{\texttt{[[],[1]]}}$.

On reconnaît l'ensemble des parties de $[\![1, 1]\!] = \{1\}$ *car* $\mathscr{P}(\{1\}) = \{\emptyset, \{1\}\}$.

(b) *Montrer que* `mystere(2)` *renvoie* `[[],[2],[1],[1,2]]`.

▶ Lorsque $n = 2$, la condition «`if n==1`» n'est pas vérifiée. Alors la fonction initialise deux listes : `Pold` égale à `mystere(1)` (car $n - 1 = 1$), donc `Pold=[[],[1]]` d'après le résultat de la question précédente, et `Pnew=[]`. Puis la fonction répète des opérations à l'aide d'une boucle `for`. Puisque `Pold` contient deux éléments (la liste vide `[]` et la liste `[1]`), `len(Pold)` est égale à 2, donc la variable `i` prend les valeurs 0 et 1.

— Pour $i = 0$, la fonction initialise la liste `L` égale à `Pold[0]=[]`, ajoute la liste `L=[]` à la liste `Pnew=[]` qui devient donc `Pnew=[[]]`, ajoute $n = 2$ à la liste `L=[]` qui devient donc `L=[2]`, et enfin ajoute la liste `L=[2]` à la liste `Pnew=[[]]` qui devient donc `Pnew=[[],[2]]`.

— Pour $i = 1$, la fonction initialise la liste `L` égale à `Pold[1]=[1]`, ajoute la liste `L=[1]` à la liste `Pnew=[[],[2]]` qui devient donc `Pnew=[[],[2],[1]]`, ajoute $n = 2$ à la liste `L=[1]` qui devient donc `L=[1,2]`, et enfin ajoute la liste `L=[1,2]` à la liste `Pnew=[[],[2],[1]]` qui devient donc `Pnew=[[],[2],[1],[1,2]]`.

Finalement, `mystere(2)` renvoie bien $\boxed{\texttt{[[],[2],[1],[1,2]]}}$.

On reconnaît l'ensemble des parties de $[\![1, 2]\!] = \{1, 2\}$ *car :*

$$\mathscr{P}\Big(\{1, 2\}\Big) = \Big\{\emptyset, \{1\}, \{2\}, \{1, 2\}\Big\}.$$

(c) *Que renvoie* `mystere(3)` *?*

▶ On raisonne comme à la question précédente. La fonction initialise la liste `Pold` égale à `mystere(2)`, donc `Pold=[[],[2],[1],[1,2]]` d'après le résultat de la question précédente. Puisque `Pold` contient quatre éléments, la variable `i` prend les valeurs 0, 1, 2 et 3.

— Pour $i=0$, la fonction initialise la liste `L` égale à `Pold[0]=[]`, ajoute la liste `L=[]` à la liste `Pnew=[]` qui devient donc `Pnew=[[]]`, ajoute $n=3$ à la liste `L=[]` qui devient donc `L=[3]`, et enfin ajoute la liste `L=[3]` à la liste `Pnew=[[]]` qui devient donc `Pnew=[[],[3]]`.

— Pour $i=1$, la fonction initialise la liste `L` égale à `Pold[1]=[2]`, ajoute la liste `L=[2]` à la liste `Pnew=[[],[3]]` qui devient donc `Pnew=[[],[3],[2]]`, ajoute $n=3$ à la liste `L=[2]` qui devient donc `L=[2,3]`, et enfin ajoute la liste `L=[2,3]` à la liste `Pnew=[[],[3],[2]]` qui devient donc `Pnew=[[],[3],[2],[2,3]]`.

— Pour $i=2$, la fonction initialise la liste `L` égale à `Pold[2]=[1]`, ajoute la liste `L=[1]` à la liste `Pnew=[[],[3],[2],[2,3]]` qui devient donc `Pnew=[[],[3],[2],[2,3],[1]]`, ajoute $n=3$ à la liste `L=[1]` qui devient donc `L=[1,3]`, et enfin ajoute la liste `L=[1,3]` à la liste `Pnew=[[],[3],[2],[2,3],[1]]` qui devient donc `Pnew=[[],[3],[2],[2,3],[1],[1,3]]`.

— De même pour $i=3$, puisque `Pold[3]=[1,2]`, la fonction ajoute la liste `[1,2]` puis la liste `[1,2,3]` à la liste `Pnew` qui devient donc `Pnew=[[],[3],[2],[2,3],[1],[1,3],[1,2],[1,2,3]]`.

Finalement, `mystere(3)` renvoie `[[],[3],[2],[2,3],[1],[1,3],[1,2],[1,2,3]]`.

On reconnaît l'ensemble des parties de $[\![1,3]\!]=\{1,2,3\}$ car :

$$\mathscr{P}\Big(\{1,2,3\}\Big)=\Big\{\emptyset,\{1\},\{2\},\{3\},\{1,2\},\{1,3\},\{2,3\},\{1,2,3\}\Big\}.$$

(d) À quoi sert la fonction `mystere` *?*

► En généralisant le raisonnement des questions précédentes, `mystere(n)` renvoie une liste modélisant $\boxed{\text{l'ensemble des parties de } [\![1,n]\!]}$.

(e) Que renvoie l'instruction `len(mystere(n))` *en fonction de l'entier $n\in\mathbb{N}^*$?*

► On sait que :

$$\operatorname{card}\Big(\mathscr{P}\Big([\![1,n]\!]\Big)\Big)=2^{\operatorname{card}([\![1,n]\!])}=2^n.$$

D'après le résultat de la question précédente, `len(mystere(n))` renvoie donc $\boxed{2^n}$.

Vérifiez la cohérence de vos résultats avec les questions (a), (b) et (c). Par exemple, puisque `mystere(1)` *renvoie* `[[],[1]]` *d'après la question (a), on vérifie bien que* `len(mystere(1))` *est égale à $2=2^1$.*

(f) En vous inspirant de la fonction `mystere` *et à l'aide de la fonction* `tester`*, écrire une fonction* `modeliser` *afin que l'instruction* `len(modeliser(n))` *renvoie la valeur de $\operatorname{card}(\mathscr{E}_n)$.*

► Par exemple :

```
def modeliser(n):
    if n==1:
        return [[],[1]]
    else:
        Pold=modeliser(n-1)
        Pnew=[]
        for i in range(len(Pold)):
            L=Pold[i]
            Pnew=Pnew+[L]
            L=L+[n]
            if tester(L):
                Pnew=Pnew+[L]
        return Pnew
```

C) Une expression sommatoire

Dans cette partie, on considère une application Φ définie sur l'ensemble des parties de $[\![1,n]\!]$ de la manière suivante : pour chaque partie $\{x_1,x_2,\dots,x_k\}\subset[\![1,n]\!]$ dont les éléments sont indicés par ordre croissant, c'est-à-dire $1\leqslant x_1<x_2<\cdots<x_k\leqslant n$, on pose :

$$\Phi\Big(\{x_1,x_2,\dots,x_k\}\Big)=\{x_1-1,x_2-2,\dots,x_k-k\}\quad(\textit{et } \Phi(\emptyset)=\emptyset \textit{ dans le cas où } k=0).$$

Par exemples, $\Phi(\{6,3,1,5\}) = \{0,1,2\}$ et $\Phi(\{4,2,8,7,3\}) = \{1,3\}$.

8. *Dans cette question, on fixe $k \in [\![0,n]\!]$ et on note $\mathscr{F}_{n,k}$ l'ensemble des parties de $[\![0,n-k]\!]$ contenant exactement k entiers.*

   (a) *Soit $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$ dont les éléments sont indicés par ordre croissant. Montrer que $\Phi(\{x_1, x_2, \ldots, x_k\}) \in \mathscr{F}_{n,k}$. Indication : prouver que $0 \leqslant x_1 - 1 < x_2 - 2 < \cdots < x_k - k \leqslant n - k$ en commençant par montrer que $x_i - i < x_{i+1} - (i+1)$ pour chaque $i \in [\![1, k-1]\!]$.*

   ▶ Soit $i \in [\![1, k-1]\!]$. On sait que $x_i < x_{i+1}$ car les éléments de $\{x_1, x_2, \ldots, x_k\}$ sont indicés par ordre croissant, et que $x_{i+1} \neq x_i + 1$ car les entiers x_i et x_{i+1} ne sont pas consécutifs. On en déduit que $x_i < x_{i+1} - 1$ puis en soustrayant par i de chaque côté de l'inégalité : $x_i - i < x_{i+1} - (i+1)$. Puisque ceci est vrai pour tout $i \in [\![1, k-1]\!]$, on a montré que :

$$x_1 - 1 < x_2 - 2 < \cdots < x_k - k.$$

   De plus, $x_1 - 1 \geqslant 0$ car $x_1 \geqslant 1$ et $x_k - k \leqslant n - k$ car $x_k \leqslant n$. Finalement, on a montré que :

$$0 \leqslant x_1 - 1 < x_2 - 2 < \cdots < x_k - k \leqslant n - k.$$

   On en déduit que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{x_1 - 1, x_2 - 2, \ldots, x_k - k\}$ contient exactement k entiers appartenant à $[\![0, n-k]\!]$, et donc que $\boxed{\Phi(\{x_1, x_2, \ldots, x_k\}) \in \mathscr{F}_{n,k}}$.

   Les inégalités strictes sont essentielles ici. Avec des inégalités larges, on montrerait seulement que $\Phi(\{x_1, x_2, \ldots, x_k\}$ est une partie de $[\![0, n-k]\!]$ contenant au plus k entiers (certains pouvant être égaux). Il est donc nécessaire d'utiliser que $\{x_1, x_2, \ldots, x_k\}$ ne contient pas d'entiers consécutifs.

   (b) *Soit $\{y_1, y_2, \ldots, y_k\} \in \mathscr{F}_{n,k}$ dont les éléments sont indicés par ordre croissant. Trouver une partie $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$ telle que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{y_1, y_2, \ldots, y_k\}$. Que peut-on en déduire pour l'application $\Phi : \mathscr{E}_{n,k} \to \mathscr{F}_{n,k}$?*

   ▶ On raisonne par analyse-synthèse.
   Analyse. On cherche $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$ telle que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{y_1, y_2, \ldots, y_k\}$. Or on sait que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{x_1 - 1, x_2 - 2, \ldots, x_k - k\}$. Il suffit donc que :

$$\begin{cases} x_1 - 1 = y_1 \\ x_2 - 2 = y_2 \\ \quad \ldots \\ x_k - k = y_k \end{cases} \iff \begin{cases} x_1 = y_1 + 1 \\ x_2 = y_2 + 2 \\ \ldots \\ x_k = y_k + k. \end{cases}$$

   Synthèse. On pose $\boxed{x_i = y_i + i}$ pour tout $i \in [\![1, k]\!]$. Vérifions que $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$.

   Attention : on a seulement prouvé que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{y_1, y_2, \ldots, y_k\}$ dans l'analyse, mais il faut vérifier que $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$ pour répondre à la question de l'énoncé.

   Soit $i \in [\![1, k-1]\!]$. On sait que $y_i < y_{i+1}$ car les éléments de $\{y_1, y_2, \ldots, y_k\}$ sont indicés par ordre croissant. En additionnant par i de chaque côté de l'inégalité, on obtient que : $y_i + i < y_{i+1} + i + 1 - 1$, donc que $x_i < x_{i+1} - 1$. Par conséquent, $x_i < x_{i+1}$ et $x_{i+1} \neq x_i + 1$. Puisque ceci est vrai pour tout $i \in [\![1, k-1]\!]$, on a montré que :

$$x_1 < x_2 < \cdots < x_k \quad \text{et} \quad \{x_1, x_2, \ldots, x_k\} \text{ ne contient pas d'entiers consécutifs.}$$

   De plus, $x_1 = y_1 + 1 \geqslant 1$ car $y_1 \geqslant 0$ et $x_k = y_k + k \leqslant n - k + k = n$ car $y_k \leqslant n - k$. Finalement, on a montré que :

$$1 \leqslant x_1 < x_2 < \cdots < x_k \leqslant n \quad \text{et} \quad \{x_1, x_2, \ldots, x_k\} \text{ ne contient pas d'entiers consécutifs.}$$

   On en déduit bien que $\boxed{\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}}$. De plus, $\boxed{\Phi(\{x_1, x_2, \ldots, x_k\}) = \{y_1, y_2, \ldots, y_k\}}$ d'après les calculs de l'analyse. Autrement dit, pour toute partie appartenant à $\mathscr{F}_{n,k}$, on a trouvé au moins un antécédent par Φ appartenant à $\mathscr{E}_{n,k}$. On en déduit que $\boxed{\Phi : \mathscr{E}_{n,k} \to \mathscr{F}_{n,k} \text{ est surjective}}$.

(c) *Montrer que l'application $\Phi : \mathscr{E}_{n,k} \to \mathscr{F}_{n,k}$ est injective.*

▶ Soient $\{x_1, x_2, \ldots, x_k\} \in \mathscr{E}_{n,k}$ et $\{x'_1, x'_2, \ldots, x'_k\} \in \mathscr{E}_{n,k}$ dont les éléments sont indicés par ordre croissant. On suppose que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \Phi(\{x'_1, x'_2, \ldots, x'_k\})$. Montrons que $\{x_1, x_2, \ldots, x_k\} = \{x'_1, x'_2, \ldots, x'_k\}$. On sait que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \{x_1-1, x_2-2, \ldots, x_k-k\}$ et $\Phi(\{x'_1, x'_2, \ldots, x'_k\}) = \{x'_1-1, x'_2-2, \ldots, x'_k-k\}$. Puisqu'on a supposé que $\Phi(\{x_1, x_2, \ldots, x_k\}) = \Phi(\{x'_1, x'_2, \ldots, x'_k\})$, on en déduit que $\{x_1-1, x_2-2, \ldots, x_k-k\} = \{x'_1-1, x'_2-2, \ldots, x'_k-k\}$. Par conséquent :

$$\begin{cases} x_1 - 1 = x'_1 - 1 \\ x_2 - 2 = x'_2 - 2 \\ \quad \ldots \\ x_k - k = x'_k - k \end{cases} \quad \text{donc} \quad \begin{cases} x_1 = x'_1 \\ x_2 = x'_2 \\ \ldots \\ x_k = x'_k. \end{cases}$$

On a bien montré que $\{x_1, x_2, \ldots, x_k\} = \{x'_1, x'_2, \ldots, x'_k\}$. Autrement dit, on a montré par l'absurde que toute partie appartenant à $\mathscr{F}_{n,k}$ admet au plus un antécédent par Φ appartenant à $\mathscr{E}_{n,k}$. On en déduit bien que $\boxed{\Phi : \mathscr{E}_{n,k} \to \mathscr{F}_{n,k} \text{ est injective}}$.

(d) *En déduire le cardinal de $\mathscr{E}_{n,k}$.*

▶ D'après les résultats des questions précédentes, l'application $\Phi : \mathscr{E}_{n,k} \to \mathscr{F}_{n,k}$ est injective et surjective, donc bijective. On en déduit que $\text{card}(\mathscr{E}_{n,k}) = \text{card}(\mathscr{F}_{n,k})$. Or, $\mathscr{F}_{n,k}$ est l'ensemble des parties de $[\![0, n-k]\!]$ contenant exactement k entiers, c'est-à-dire l'ensemble des k-combinaisons d'éléments de $[\![0, n-k]\!]$. Par conséquent, il y en a :

$$\text{card}\Big(\mathscr{E}_{n,k}\Big) = \text{card}\Big(\mathscr{F}_{n,k}\Big) = \binom{\text{card}\Big([\![0, n-k]\!]\Big)}{k} = \boxed{\binom{n-k+1}{k}}.$$

9. *Démontrer que :*

$$\text{card}\Big(\mathscr{E}_n\Big) = \sum_{k=0}^{n} \binom{n-k+1}{k}.$$

▶ D'après l'énoncé, on a $\mathscr{E}_n = \mathscr{E}_{n,0} \cup \mathscr{E}_{n,1} \cup \mathscr{E}_{n,2} \cup \cdots \cup \mathscr{E}_{n,n} = \bigcup_{k=0}^{n} \mathscr{E}_{n,k}$. De plus, les sous-ensembles $\mathscr{E}_{n,0}, \mathscr{E}_{n,1}, \mathscr{E}_{n,2}, \ldots, \mathscr{E}_{n,n}$ sont deux à deux disjoints (car une partie de $[\![1, n]\!]$ ne contient qu'un nombre fixé d'entiers). Par conséquent :

$$\begin{aligned} \text{card}\Big(\mathscr{E}_n\Big) &= \text{card}\left(\bigcup_{k=0}^{n} \mathscr{E}_{n,k}\right) \\ &= \sum_{k=0}^{n} \text{card}\Big(\mathscr{E}_{n,k}\Big) \quad \text{car les sous-ensembles sont deux à deux disjoints} \\ &= \boxed{\sum_{k=0}^{n} \binom{n-k+1}{k}} \quad \text{d'après le résultat de la question précédente.} \end{aligned}$$

N'oubliez pas de préciser que les ensembles sont deux à deux disjoints pour pouvoir utiliser la propriété que le cardinal d'une union est égal à la somme des cardinaux (cette propriété est fausse en général).

D) Calcul numérique

Dans cette partie, on suppose le résultat de la question 9.

10. *Écrire une fonction* `fact` *qui prend en argument un entier $m \in \mathbb{N}$, puis qui renvoie la valeur de $m!$.*

▶ Par exemple :

```
def fact(m):
    P=1
    for i in range(1,m+1):
        P=P*i
    return P
```

11. *Écrire une fonction* `coeffbi` *qui prend en arguments deux entiers $j \in \mathbb{Z}$ et $m \in \mathbb{N}$, puis qui renvoie la valeur de $\binom{m}{j}$.*

▶ Par exemple :

```
def coeffbi(j,m):
    if j<0 or j>m:
        return 0
    else:
        return fact(m)/(fact(j)*fact(m-j))
```

N'oubliez pas de distinguer les cas où $j \notin [\![0, m]\!]$ pour éviter les erreurs dans la question suivante.

12. *Écrire une fonction* `cardE` *qui prend en argument l'entier $n \in \mathbb{N}^*$, puis qui renvoie la valeur de $\mathrm{card}(\mathscr{E}_n)$.*

▶ Par exemple :

```
def cardE(n):
    S=0
    for k in range(n+1):
        S=S+coeffbi(k,n-k+1)
    return S
```

Remarquez que pour les dernières valeurs de k (par exemple pour $k = n$), on a $k > n - k + 1$ et donc $\binom{n-k+1}{k} = 0$. Ainsi, il faut bien distinguer ce cas dans la question précédente pour éviter les erreurs.

E) Résolution explicite

Pour chaque $n \in \mathbb{N}^*$, on pose $u_n = \mathrm{card}(\mathscr{E}_n)$ et on note $\mathscr{A}_n$ le sous-ensemble de $\mathscr{E}_n$ des parties contenant l'entier n. Ainsi :

$$\mathscr{A}_3 = \Big\{\{3\}, \{1, 3\}\Big\} \quad \text{et} \quad \overline{\mathscr{A}_3} = \mathscr{E}_3 \setminus \mathscr{A}_3 = \Big\{\emptyset, \{1\}, \{2\}\Big\}.$$

13. *Dans cette question, on fixe $n \geqslant 3$.*

   (a) *Justifier que $\mathrm{card}(\mathscr{A}_n) = u_{n-2}$.*

   ▶ Les parties appartenant à $\mathscr{A}_n$ sont de la forme $\{x_1, x_2, \ldots, x_{k-1}, x_k = n\}$ où :

$$1 \leqslant x_1 < x_2 < \cdots < x_{k-1} < x_k = n \leqslant n.$$

Puisque x_{k-1} et $x_k = n$ ne sont pas consécutifs, on sait que $x_k = n \neq x_{k-1} + 1$, donc que $x_{k-1} < n - 1$. Par conséquent :

$$1 \leqslant x_1 < x_2 < \cdots < x_{k-1} \leqslant n - 2.$$

On en déduit que $\{x_1, x_2, \ldots, x_{k-1}\}$ est une partie de $[\![1, n-2]\!]$ ne contenant pas d'entiers consécutifs, c'est-à-dire $\{x_1, x_2, \ldots, x_{k-1}\} \in \mathscr{E}_{n-2}$. Finalement, chaque partie appartenant à $\mathscr{A}_n$ est de la forme :

$$\underbrace{\{x_1, x_2, \ldots, x_{k-1}, x_k = n\}}_{\in \mathscr{A}_n} = \underbrace{\{x_1, x_2, \ldots, x_{k-1}\}}_{\in \mathscr{E}_{n-2}} \cup \{n\}.$$

Il y a donc autant de parties appartenant à $\mathscr{A}_n$ que de parties appartenant à $\mathscr{E}_{n-2}$, par conséquent :

$$\boxed{\mathrm{card}\Big(\mathscr{A}_n\Big) = \mathrm{card}(\mathscr{E}_{n-2}) = u_{n-2}}.$$

(b) Déterminer une expression similaire pour le cardinal du complémentaire de $\mathscr{A}_n$ dans $\mathscr{E}_n$.

▶ Les parties de $\overline{\mathscr{A}_n} = \mathscr{E}_n \setminus \mathscr{A}_n$ sont de la forme $\{x_1, x_2, \dots, x_k\}$ où :

$$1 \leqslant x_1 < x_2 < \cdots < x_k \leqslant n.$$

Puisque $\{x_1, x_2, \dots, x_k\}$ ne contient pas l'entier n, on sait que $x_k \neq n$, donc que $x_k < n$. Par conséquent :

$$1 \leqslant x_1 < x_2 < \cdots < x_k \leqslant n-1.$$

On en déduit que $\{x_1, x_2, \dots, x_k\}$ est une partie de $[\![1, n-1]\!]$ ne contenant pas d'entiers consécutifs, c'est-à-dire $\{x_1, x_2, \dots, x_k\} \in \mathscr{E}_{n-1}$. Il y a donc autant de parties appartenant à $\overline{\mathscr{A}_n}$ que de parties appartenant à $\mathscr{E}_{n-1}$, par conséquent :

$$\boxed{\operatorname{card}\left(\overline{\mathscr{A}_n}\right) = \operatorname{card}\left(\mathscr{E}_{n-1}\right) = u_{n-1}}.$$

(c) Montrer que $u_n = u_{n-1} + u_{n-2}$.

▶ On a par définition du complémentaire :

$$\mathscr{E}_n = \mathscr{A}_n \cup \overline{\mathscr{A}_n} \quad \text{et} \quad \mathscr{A}_n \cap \overline{\mathscr{A}_n} = \emptyset.$$

Par conséquent :

$$\begin{aligned} u_n &= \operatorname{card}\left(\mathscr{E}_n\right) \\ &= \operatorname{card}\left(\mathscr{A}_n \cup \overline{\mathscr{A}_n}\right) \\ &= \operatorname{card}\left(\mathscr{A}_n\right) + \operatorname{card}\left(\overline{\mathscr{A}_n}\right) \quad \text{car } \mathscr{A}_n \cap \overline{\mathscr{A}_n} = \emptyset \\ &= u_{n-2} + u_{n-1} \quad \text{d'après les résultats des questions précédentes.} \end{aligned}$$

On a bien montré que $\boxed{u_n = u_{n-1} + u_{n-2}}$.

14. *En déduire le terme général de la suite $(\operatorname{card}(\mathscr{E}_n) = u_n)_{n \geqslant 1}$. Pour simplifier les calculs, on pourra poser la constante $\varphi = (1 + \sqrt{5})/2$.*

▶ Quitte à faire un décalage d'indice, on a montré à la question précédente que :

$$\forall n \geqslant 1, \quad u_{n+2} = u_{n+1} + u_n.$$

On reconnaît une suite récurrente linéaire d'ordre deux d'équation caractéristique :

$$q^2 = q + 1 \iff q^2 - q - 1 = 0.$$

On reconnaît une équation de degré deux à coefficients réels dont le discriminant est égal à :

$$\Delta = (-1)^2 - 4 \times 1 \times (-1) = 5 > 0.$$

L'équation caractéristique admet donc deux solutions réelles distinctes :

$$q_1 = \frac{-(-1) + \sqrt{\Delta}}{2 \times 1} = \frac{1 + \sqrt{5}}{2} = \varphi \quad \text{et} \quad q_2 = \frac{1 - \sqrt{5}}{2} = 1 - \varphi.$$

Pensez à utiliser la somme des solutions (égale à $q_1 + q_2 = -(-1) = 1$) ou le produit des solutions (égal à $q_1 q_2 = -1$) pour simplifier l'expression de la deuxième solution : $q_2 = 1 - q_1 = -1/q_1$.

On sait qu'il existe deux constantes $(\lambda_1, \lambda_2) \in \mathbb{R}^2$ telles que :

$$\forall n \geqslant 1, \quad u_n = \lambda_1 q_1^{n-1} + \lambda_2 q_2^{n-1} = \lambda_1 \varphi^{n-1} + \lambda_2 (1-\varphi)^{n-1}.$$

De plus, on sait d'après la question 1 que :

$$u_1 = \mathrm{card}\Big(\mathscr{E}_1\Big) = 2 \quad \text{et} \quad u_2 = \mathrm{card}\Big(\mathscr{E}_2\Big) = 3.$$

Donc :

$$\begin{aligned}
&\begin{cases} 2 = u_1 = \lambda_1 \underbrace{\varphi^{1-1}}_{=\varphi^0=1} + \lambda_2 \underbrace{(1-\varphi)^{1-1}}_{=(1-\varphi)^0=1} = \lambda_1 + \lambda_2 \\ 3 = u_2 = \lambda_1 \underbrace{\varphi^{2-1}}_{=\varphi^1=\varphi} + \lambda_2 \underbrace{(1-\varphi)^{2-1}}_{=(1-\varphi)^1=1-\varphi} = \lambda_1 \varphi + \lambda_2(1-\varphi) \end{cases} \\
\iff &\begin{cases} \lambda_1 = 2 - \lambda_2 \\ 3 = (2-\lambda_2)\varphi + \lambda_2(1-\varphi) = 2\varphi + \lambda_2(1-2\varphi) \end{cases} \\
\iff &\begin{cases} \lambda_2 = \dfrac{3-2\varphi}{1-2\varphi} \\ \lambda_1 = 2 - \dfrac{3-2\varphi}{1-2\varphi} = \dfrac{-1-2\varphi}{1-2\varphi} \end{cases} \\
\iff &\begin{cases} \lambda_2 = \dfrac{2-\sqrt{5}}{-\sqrt{5}} = 1 - \frac{2}{5}\sqrt{5} \\ \lambda_1 = \dfrac{-2-\sqrt{5}}{-\sqrt{5}} = 1 + \frac{2}{5}\sqrt{5} \end{cases} \quad \text{car } \varphi = \dfrac{1+\sqrt{5}}{2}.
\end{aligned}$$

Finalement, on obtient que :

$$\forall n \geqslant 1, \quad \mathrm{card}\Big(\mathscr{E}_n\Big) = u_n = \boxed{\left(1 + \frac{2}{5}\sqrt{5}\right)\left(\frac{1+\sqrt{5}}{2}\right)^{n-1} + \left(1 - \frac{2}{5}\sqrt{5}\right)\left(\frac{1-\sqrt{5}}{2}\right)^{n-1}}.$$