DS n°1 : corrigé

1°)

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
f_n	0	1	1	2	3	5	8	13	21	34	55	89	144	233	377	610

2°) * Soit $u=(u_n)\in F$, $v=(v_n)\in F$ et $\lambda\in\mathbb{R}$

Mq $\underline{\lambda u+v\in F}$

$$\forall n\in\mathbb{N}\quad (\lambda u+v)_{n+2}=\lambda u_{n+2}+v_{n+2}$$
$$=\lambda(u_{n+1}+u_n)+v_{n+1}+v_n$$
$$=[\lambda u_{n+1}+v_{n+1}]+[\lambda u_n+v_n]$$
$$=(\lambda u+v)_{n+1}+(\lambda u+v)_n$$

donc $\underline{\lambda u+v\in F}$

* Notons $\theta=(0)$ (la suite toute nulle)

$$\forall n\in\mathbb{N}\quad \theta_{n+2}=0=0+0=\theta_{n+1}+\theta_n$$

donc $\theta\in F$ et $\underline{F\neq\emptyset}$

d° $\boxed{F \text{ sev de } E \text{ et donc } F\ \mathbb{R}\text{-ev.}}$

3°) Montrons par récurrence **double** que $u_n = v_n$:

$n=0$ et $n=1$: oui

Supposons $u_n = v_n$ et $u_{n+1} = v_{n+1}$ pour un n donné.

Alors $*$ $u_{n+1} = v_{n+1}$

$*$ $u_{n+2} = u_{n+1} + u_n = v_{n+1} + v_n$

$= v_{n+2}$ car $(u,v) \in F^2$

d'où $\boxed{\forall n \in \mathbb{N} \quad u_n = v_n}$

4°) $\Rightarrow$] Si $(r^n) \in F$, pour $n=0$:

$$r^{0+2} = r^{0+1} + r^0 \Rightarrow \underline{r^2 = r+1}$$

$\Leftarrow$] Si $r^2 = r+1$, $\forall n \in \mathbb{N}$:

$$r^{n+2} = r^n(r+1) = r^{n+1} + r^n \therefore (r^n) \in F$$

d'où $\boxed{(r^n) \in F \Leftrightarrow r^2 = r+1}$

5°) $\Delta = 5$ d'où $\boxed{r_1 = \frac{1-\sqrt{5}}{2} \text{ et } r_2 = \frac{1+\sqrt{5}}{2}}$

6°) On résoud le système (S) $\begin{cases} \alpha + \beta = 0 & (n=0) \\ \alpha r_1 + \beta r_2 = 1 & (n=1) \end{cases}$

$$(S) \iff \begin{cases} \beta = -\alpha \\ \alpha(r_1 - r_2) = 1 \end{cases} \iff \begin{cases} \alpha = \dfrac{1}{r_1 - r_2} = \dfrac{-1}{\sqrt{5}} \\ \beta = \dfrac{1}{\sqrt{5}} \end{cases}$$

③

Avec le 3°) :

d'où $$\boxed{\forall n \in \mathbb{N},\ f_n = -\frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n + \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n}$$

7°) $$\frac{q^n + r^n}{r^n} = 1 + \left(\frac{q}{r}\right)^n \quad \text{comme } \left|\frac{q}{r}\right| < 1,\ \left(\frac{q}{r}\right)^n \xrightarrow[n\to\infty]{} 0$$

$$\xrightarrow[n\to\infty]{} 1$$

d'où $\boxed{q^n + r^n \sim q^n}$

8) On fait de m̂ : $$\frac{f_n}{\frac{1}{\sqrt{5}} r_2^n} = 1 + \left(\frac{r_1}{r_2}\right)^n$$

Or $$\left|\frac{r_1}{r_2}\right| = \frac{\sqrt{5}-1}{\sqrt{5}+1} < \frac{\sqrt{5}}{\sqrt{5}+1} < 1, \text{ donc } \left(\frac{r_1}{r_2}\right)^n \xrightarrow[n\to\infty]{} 0$$

d'où $$\boxed{f_n \underset{+\infty}{\sim} \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n}$$

9) $$0 < \frac{1}{f_n} \sim \sqrt{5}\left(\frac{2}{1+\sqrt{5}}\right)^n, \text{ Posons } q = \frac{1}{r_2} = \frac{2}{1+\sqrt{5}}$$

On a $0 < q < \dfrac{2}{1+2} = \dfrac{2}{3} < 1 \quad (\sqrt{5} > 2)$

Cqs $(\sum \sqrt{5}\, q^n)$ cvg, $\frac{1}{\sqrt{5}} q^n \geq 0$ d'où par T.C. : $\boxed{\left(\sum \frac{1}{f_n}\right) \text{cvg}}$

10°) On a $\forall n \in [\![0,5]\!] : \underline{d_n = 1}$

<u>Conjecture</u> : $\underline{\forall n \in \mathbb{N} : d_n = 1}$

11°) $\Rightarrow]$ si $\delta | f_n$ et $\delta | f_{n+1}$

<u>alors</u> $\delta | f_n$ et $\delta | f_{n+1} - f_n = f_{n-1}$

$\Leftarrow]$ si $\delta | f_n$ et $\delta | f_{n-1}$

<u>alors</u> $\delta | f_n$ et $\delta | f_n + f_{n-1} = f_{n+1}$

d'où $\boxed{\text{On a l'équivalence}}$

cqs $\forall n \geq 3 \quad d_n = d_{n-1}$ et par récurrence descendante :

$\forall n \geq 3 \quad d_n = d_2 = 1$ d'où $\boxed{\forall n \in \mathbb{N} \quad d_n = 1}$

12°)

```
def f(n):
    if n==0:
        return 0
    elif n==1:
        return 1
    else:
        return (f(n-1) + f(n-2))
```

Avec un ordinateur d'aujourd'hui il faudrait des millénaires pour calculer $f(100)$!...
Ou bien écrire une fonction itérative.

13') a) $Tr(\Pi) = \frac{1}{2} + \frac{1}{2} = 1$

$$rg(\Pi) = rg\begin{pmatrix}1&1\\1&0\\1&0\\1&1\end{pmatrix} \quad (C_3 = C_2 \text{ et } C_4 = C_1)$$

$$= rg\begin{pmatrix}1&0\\1&-1\\1&-1\\1&0\end{pmatrix} = 2 < 4 \text{ donc } \det\Pi = 0$$

cl° $\boxed{tr(\Pi)=1,\ rg(\Pi)=2,\ \det(\Pi)=0 \text{ et } \Pi \text{ non inversible}}$

b)
$$\begin{array}{ccc} & & \begin{pmatrix}1&1&1&1\\1&0&0&1\\1&0&0&1\\1&1&1&1\end{pmatrix} \\ \Pi^2 = \frac{1}{4} & \begin{pmatrix}1&1&1&1\\1&0&0&1\\1&0&0&1\\1&1&1&1\end{pmatrix} & \begin{pmatrix}4&2&2&4\\2&2&2&2\\2&2&2&2\\4&2&2&4\end{pmatrix}\end{array} \quad \text{et } \Pi^3 = \frac{1}{8}\begin{pmatrix}12&8&8&12\\8&4&4&8\\8&4&4&8\\12&8&8&12\end{pmatrix}$$

cqs $\Pi^2 = \frac{1}{2}\begin{pmatrix}2&1&1&2\\1&1&1&1\\1&1&1&1\\2&1&1&2\end{pmatrix}$ et $\Pi^3 = \frac{1}{2}\begin{pmatrix}3&2&2&3\\2&1&1&2\\2&1&1&2\\3&2&2&3\end{pmatrix}$

On en déduit $\Pi + \Pi^2 = \frac{1}{2}\begin{pmatrix}3&2&2&3\\2&1&1&2\\2&1&1&2\\3&2&2&3\end{pmatrix} = \Pi^3$

cl° $\boxed{\Pi + \Pi^2 = \Pi^3}$

c) c'est vrai pour $n = 1, 2, 3$.
Supposons le vrai pour $n \geq 1$

$$\Pi^{n+1} = \Pi\ \Pi^n = \Pi(a_n\Pi + b_n\Pi^2) = a_n\Pi^2 + b_n\Pi^3$$

donc $\Pi^{n+1} = a_n \Pi^2 + b_n (\Pi + \Pi^2) = \underbrace{b_n}_{\in \mathbb{R}} \Pi + \underbrace{(a_n + b_n)}_{\in \mathbb{R}} \Pi^2$

cqs c'est encore vrai pour $n+1$

De plus par unicité $\underline{a_{n+1} = b_n}$ et $\underline{b_{n+1} = a_n + b_n}$

d'où $\boxed{\forall n \geq 1 \ \exists (a_n, b_n) \in \mathbb{R}^2 \mid \Pi^n = a_n \Pi + b_n \Pi^2}$

$\Pi^1 = 1.\Pi + 0.\Pi^2$ donc $\underline{a_1 = 1}$ et $\underline{b_1 = 0}$

$\Pi^2 = 0.\Pi + 1.\Pi^2$ donc $\underline{a_2 = 0}$ et $\underline{b_2 = 1}$

d) $\forall n \geq 1 \quad b_{n+2} = a_{n+1} + b_{n+1} = b_n + b_{n+1}$

vérif pour $n = 0$: $b_2 = 1 = 1 + 0 = b_0 + b_1$

d'où $\boxed{(b_n) \in F}$

$\forall n \geq 2 \quad a_{n+2} = b_{n+1} = b_n + b_{n-1}$ car $(b_n) \in F$

$= a_{n+1} + a_n$

vérif pour $n = 0 \quad a_2 = 0 = 1 - 1 = a_1 + a_0$

$n = 1 \quad a_3 = b_2 = 1 = 0 + 1 = a_2 + a_1$

d'où $\boxed{(a_n) \in F}$

n	0	1	2	3	4	5	6	7	8	9	10
f_n	0	1	1	2	3	5	8	13	21	34	55
a_n	-1	1	0	1	1	2	3	5	8	13	21
b_n	1	0	1	1	2	3	5	8	13	21	34

d° $\boxed{\forall n \geq 2 \quad a_n = f_{n-2} \quad \text{et} \quad b_n = f_{n-1}}$

14) a) * $k=1$

On montre facilement par récurrence double que $\forall n \geq 1$ $f_n > 0$ donc $\forall n \geq 2$ $f_n = f_{n+1} - \underbrace{f_{n-1}}_{>0} < f_{n+1}$

donc H_1 vraie

* supposons le lemme vrai pour $k \geq 1$

$$\underbrace{f_{n_1} + \cdots + f_{n_k}} + f_{n_{k+1}}$$

$$< f_{n_k+1} + f_{n_{k+1}} \quad \text{par hyp. de récurrence}$$

Or $n_{k+1} \geq n_k + 2$ donc $n_k + 1 \leq n_{k+1} - 1$ d'où

$$f_{n_1} + \cdots + f_{n_k} + f_{n_{k+1}} < f_{n_{k+1}-1} + f_{n_{k+1}} = f_{n_{k+1}+1}$$

donc H_{k+1} vraie

d'o $\boxed{\text{Le lemme est vrai.}}$ (8)

b) Analyse : si $N = f_{n_1} + \cdots + f_{n_k}$

On a donc $N \geqslant 0 + \cdots + 0 + f_{n_k} = f_{n_k}$ et avec

le lemme $N < f_{n_k+1}$

d'o $\boxed{f_{n_k} \leqslant N < f_{n_k+1}}$

c) $\boxed{1 = f_2}$ $\boxed{2 = f_3}$

Avec le 1°) a) on a $\underset{f_8}{\underset{\|}{21}} \leqslant 25 < \underset{f_9}{\underset{\|}{34}}$

donc $25 = 21 + 4$ et $\underset{f_4}{\underset{\|}{3}} \leqslant 4 < \underset{f_5}{\underset{\|}{5}}$

$= 21 + 3 + 1$

cqs $\boxed{25 = f_8 + f_4 + f_2}$ (et 1, 4, 8 non consécutifs)

Idem $100 = f_{11} + 11 = f_{11} + f_6 + \underbrace{3}_{= f_4}$

cqs $\boxed{100 = f_{11} + f_6 + f_4}$

9

d) Récurrence forte sur N : H_N

H_1 est vrai

Supposons H_n vraie pour $n \leq N$, on a vu au 14 a) que $(f_n)_{n \geqslant 2}$ était strictement croissante donc le n_k du lemme existe et est unique.

On a donc $N = f_{n_k} + n$ avec $n = N - f_{n_k} < N$

1er cas $n = 0$ alors $N = f_{n_k}$ est la déc. de Zeckendorf.

2e cas $n \geqslant 1$: on applique l'hypothèse de réc. à n :

$\exists ! (n_1, \dots, n_{k-1}) \in \mathbb{N}^{k-1} \mid \forall i : n_{i+1} \geqslant n_i + 2$ et

$n = f_{n_1} + \dots + f_{n_{k-1}}$ d'où $N = f_{n_1} + \dots + f_{n_{k-1}} + f_{n_k}$

d'où $\boxed{\text{le théorème est vrai}}$

e)
```
def zec(N):
    if N==1:
        return [2]
    else:
        c=0
        while f(c)<=N:
            c=c+1
        if N==f(c-1): #ici k=c-1
            return [c-1]
        else:
            return [c-1]+zec(N-a(c-1))
```

```
from time import *

def f(n):
    if n<=1:
        return n
    else:
        return f(n-1)+f(n-2)

def temps(n): # temps pour renvoyer f(n)
    if n not in dico:
        aux=temps(n-1)+temps(n-2)
        dico[n]=aux
    return dico[n]


def zec(N):
    if N==1:
        return [2]
    else:
        c=0
        while f(c)<=N:
            c=c+1
        if N==f(c-1):
            return [c-1]
        else:
            return [c-1]+zec(N-f(c-1))

def verifZec(L):
    s=0
    k=len(L)
    for i in range(k):
        s=s+f(L[i])
    return s

n0=35
t0=time()
f(n0)
print(time()-t0)
t0=time()
f(n0+1)
print(time()-t0)

dico={35:3.8087830543518066,36:6.219388723373413} #n0=35

#>>> temps(100)/3600/24/365/1000
#------>4667.188737063703 millénaires pour renvoyer f(100)!!!!
```