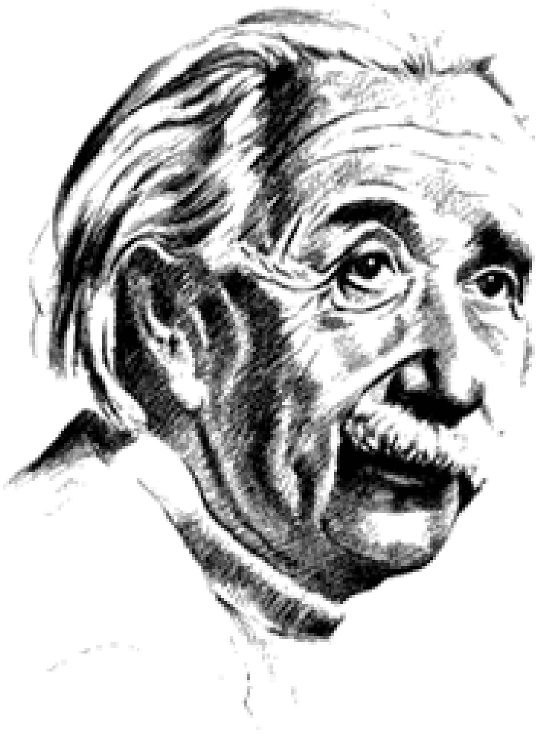
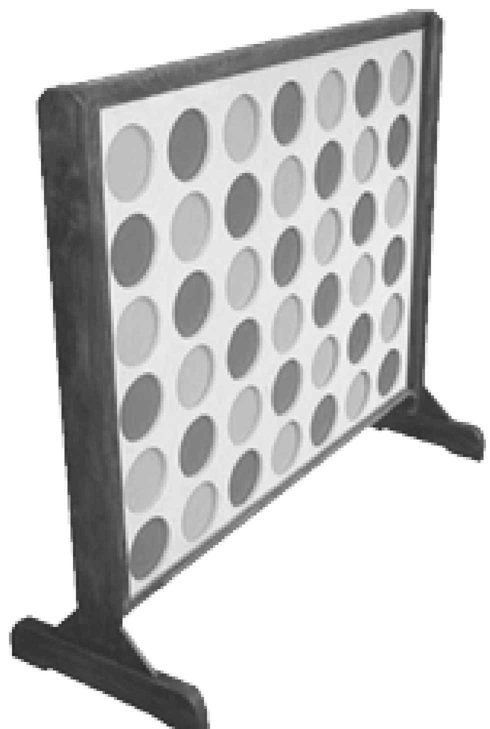
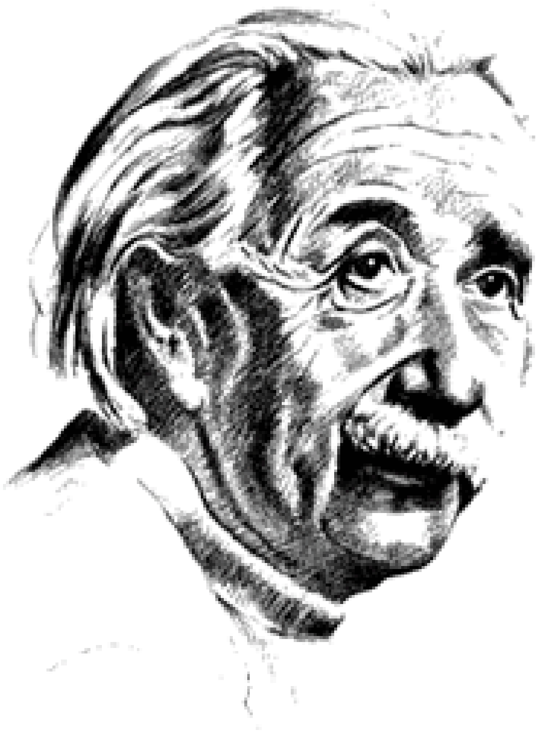
Le puissance 4





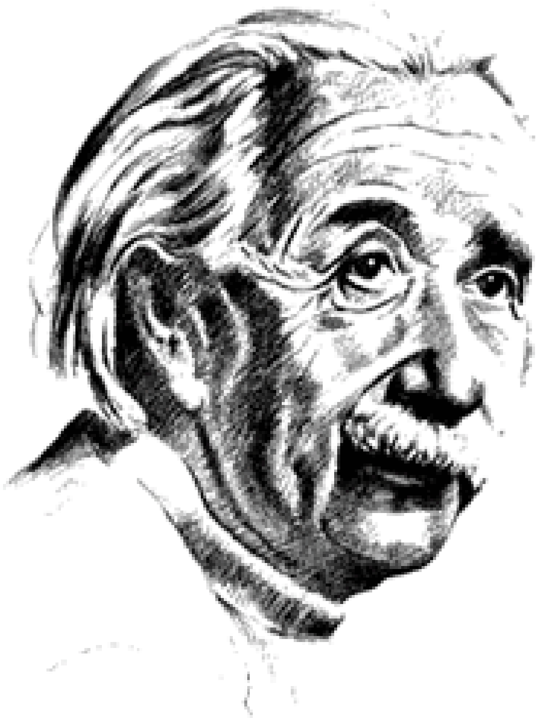
Début de notre réflexion

Comment transformer les règles du *puissance 4* sous forme mathématiques ?

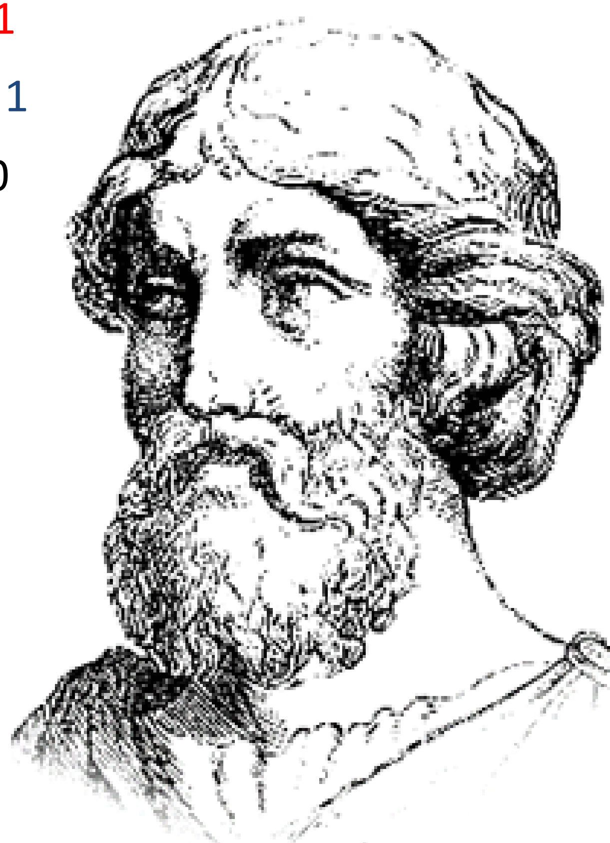


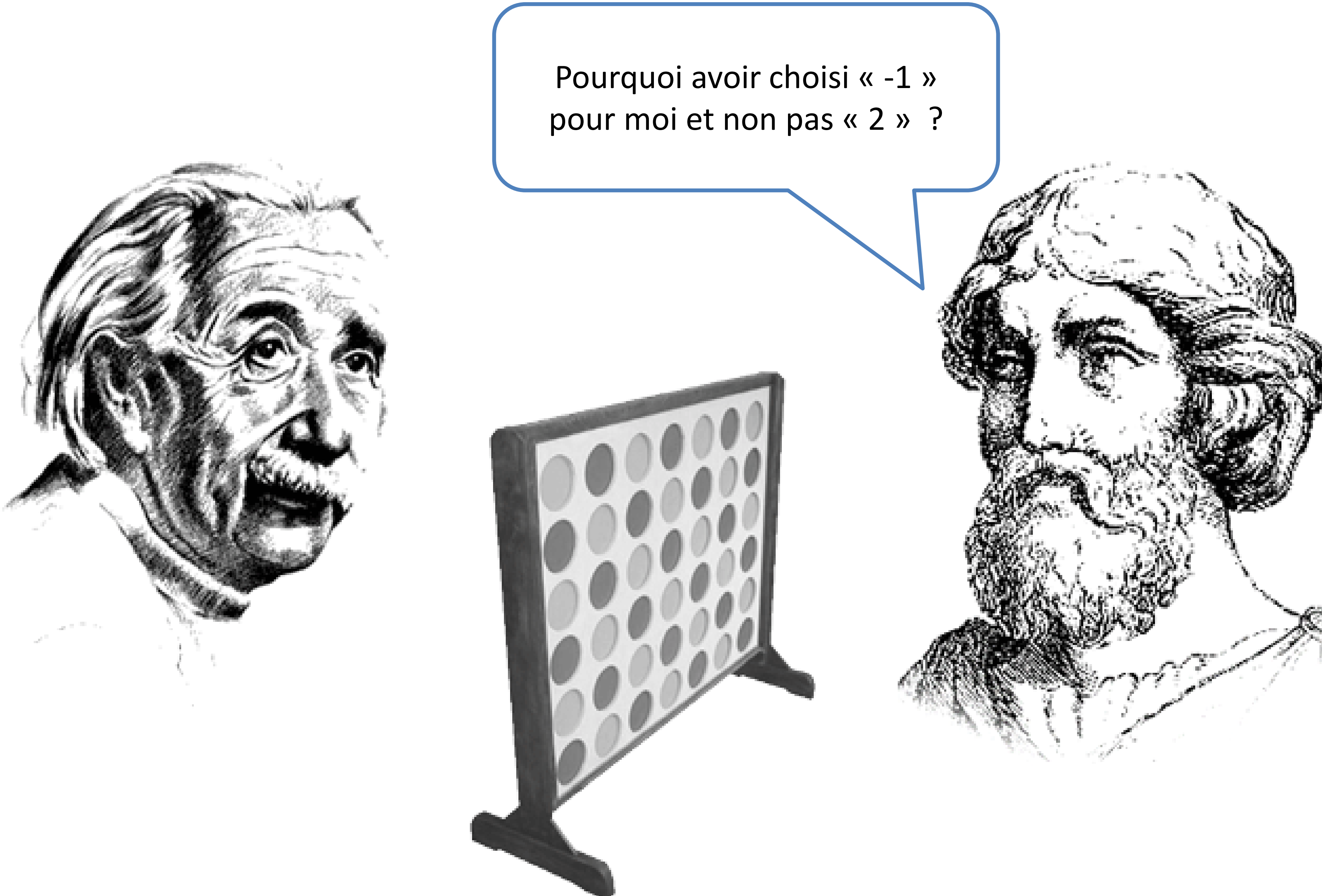
Mathématisation des règles

- Le plateau sera une matrice
- Pour Einstein (joueur 1) = 1
- Pour Pythagore (joueur 2) = - 1
- Pour une case vide = 0



$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$




Pourquoi avoir choisi « -1 »
pour moi et non pas « 2 » ?

Exemple :

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 1 & -1 & 0 & 1 \end{pmatrix}$$

$-1 + 1 + -1 + 1 + 1 + -1 + 1 = 1$

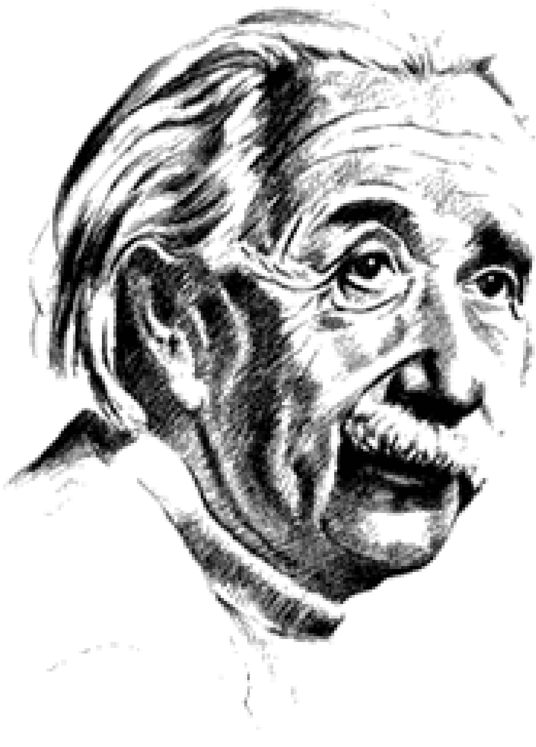
C'est donc à Pythagore (joueur 2) de jouer.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 1 & -1 & 0 & 1 \end{pmatrix}$$

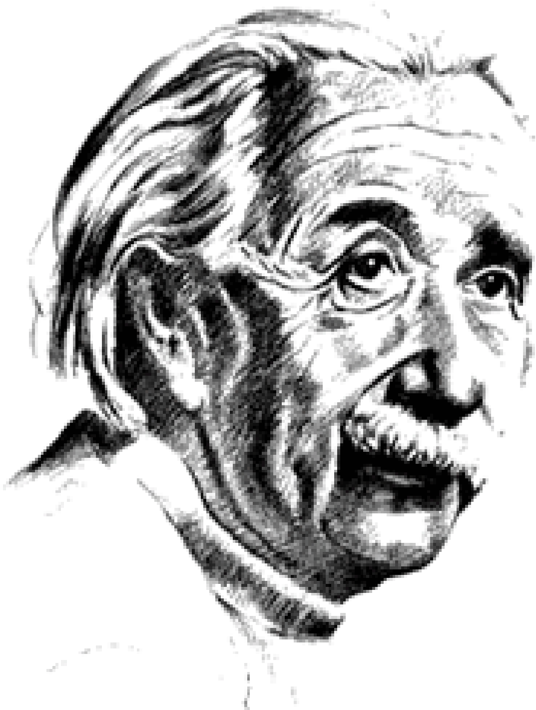
$-1 + 1 + -1 + -1 + 1 + 1 + -1 + 1 = 0$

C'est donc à Einstein (joueur 1) de jouer.

J’ai compris. Commençons la partie.



$$\begin{pmatrix} 0 & 0 & 0 & \color{red}{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$



Pourquoi mon jeton ne descend-t-il pas ?

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Nouvel opérateur

Nom : Gravite

Symbol : 

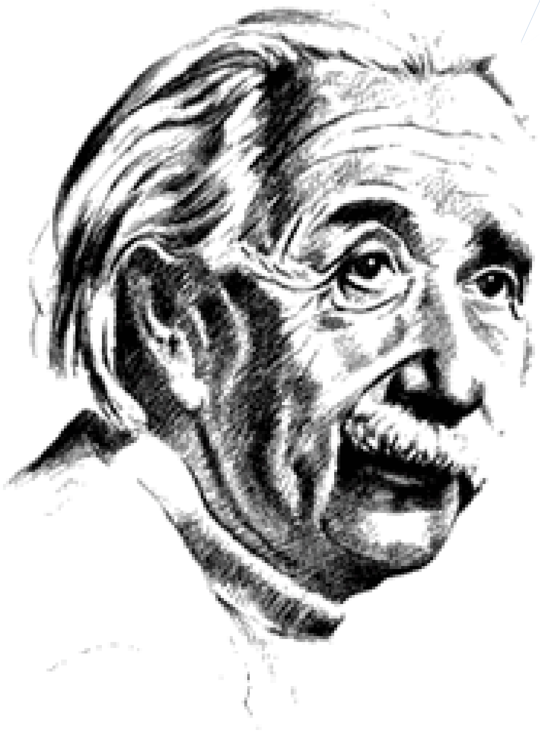
Définition :

Remplace, dans une matrice N x M, les « 0 » de la première ligne par les chiffres différents de « 0 », d'une matrice ligne M colonnes.

Puis on applique un principe de **« gravité »** à la première matrice.

Propriété de la gravité dans une matrice :

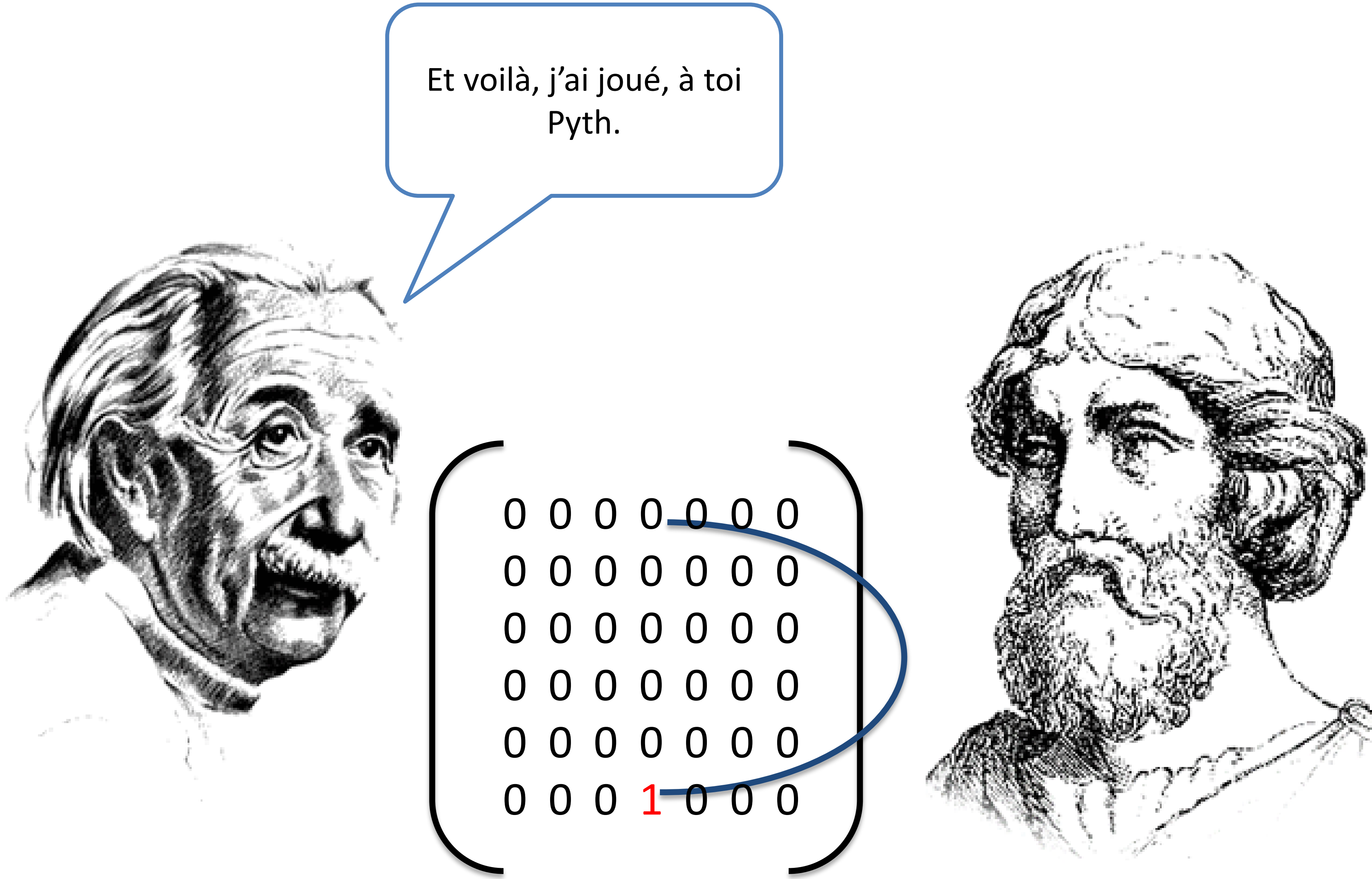
En commençant par le bas de chaque colonne, on remplace le premier « 0 » par le premier chiffre rencontré en remontant cette colonne. Et ainsi de suite jusqu’à la fin de celle-ci. Le même principe est utilisé pour toutes les colonnes.



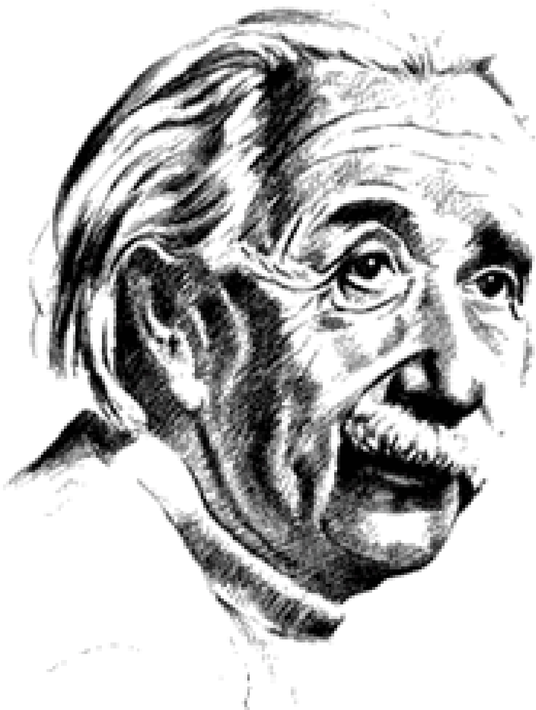
$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\Downarrow$$

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$


Et voilà, j’ai joué, à toi Pyth.
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 1 0 0 0

Plus tard …



Mais comment savoir si j’ai gagné mathématiquement ?

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 & 0 \\ -1 & 1 & 1 & 1 & -1 & -1 & 0 \end{pmatrix}$$

Les vérifications

Pour vérifier les lignes :

Nous multiplions notre matrice par une matrice ligne remplie de « 0 », sauf au numéro de la ligne que nous voulons extraire où nous mettons un « 1 ». Puis nous multiplions par une matrice colonne remplie de « 1 ».

Exemple :

Pour vérifier la deuxième ligne de cette matrice.

Donc obtenir d + e + f

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

Etape 1 :

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

$$\times$$

$$\begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} d & e & f \end{pmatrix}$$

Etape 2 :

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\times$$

$$\begin{pmatrix} d & e & f \end{pmatrix} \qquad \begin{pmatrix} d+e+f \end{pmatrix}$$

Les vérifications

Pour vérifier les colonnes :

Nous multiplions par une matrice colonne remplie de « 0 », sauf au numéro de la colonne que nous voulons extraire où nous mettons un « 1 ». Puis nous multiplions notre matrice par une matrice ligne remplie de « 1 ».

Exemple :

Pour vérifier la deuxième colonne de cette matrice.

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

Donc obtenir b + e + h

Etape 1 :

$$\times \quad \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} \qquad \begin{bmatrix} b \\ e \\ h \end{bmatrix}$$

Etape 2 :

$$\times \quad \begin{bmatrix} b \\ e \\ h \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \qquad \begin{bmatrix} b+e+h \end{bmatrix}$$

Les vérifications

Pour vérifier la diagonale principale :

Nous utilisons un outil mathématique existant : La trace.

Soit A notre matrice et $a_{i\,j}$ ses coefficients tels que :

$$A = (a_{i\,j})_{1 \leq i,j \leq n}$$

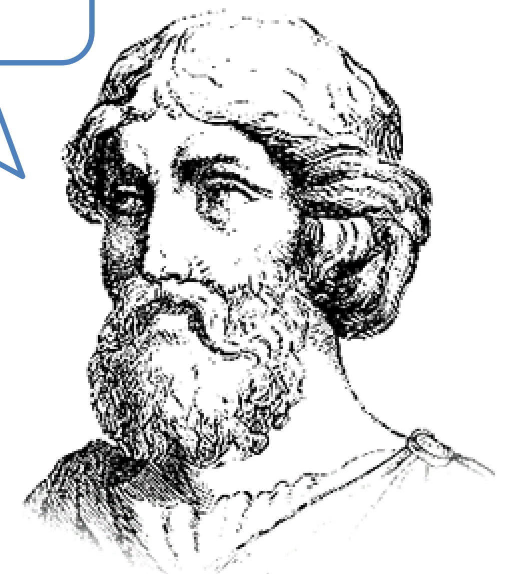
La trace est la somme des coefficients de la diagonale principale, noté :

$$\mathrm{Tr}(A) = \sum_{i=1}^{n} a_{i\,i}$$

Exemple :

$$\mathrm{Tr}\left(\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}\right) = a+e+i$$

Et pour l'autre diagonale ?



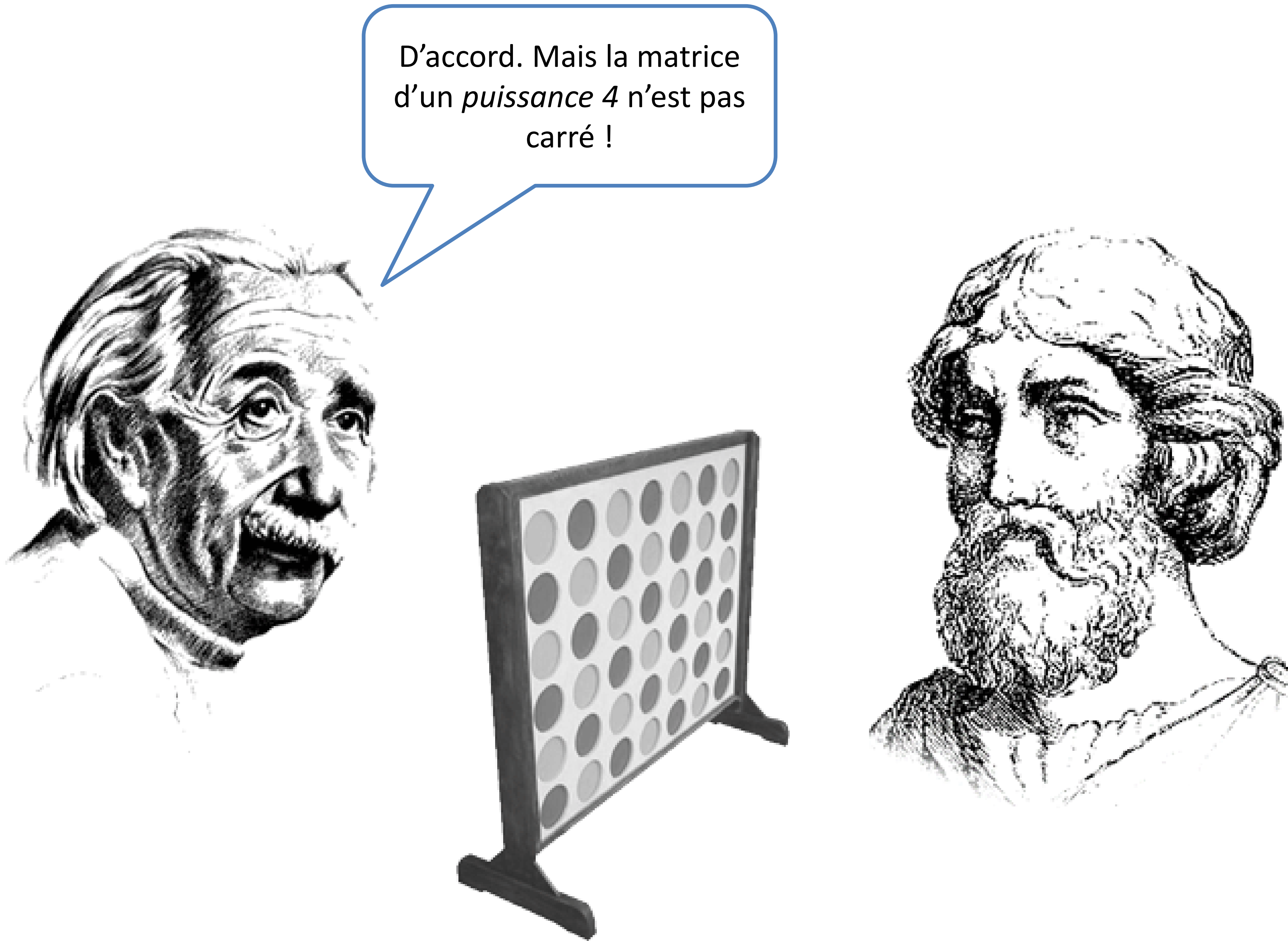
Les vérifications

Pour vérifier la diagonale secondaire :

Nous faisons la « symétrie » de la matrice par rapport à une droite placée verticalement au milieu de la matrice.

$$\begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & p & q \end{pmatrix} \Longrightarrow \begin{pmatrix} d & c & b & a \\ h & g & f & e \\ l & k & j & i \\ q & p & n & m \end{pmatrix}$$

Puis nous refaisons la trace avec cette nouvelle matrice.


D’accord. Mais la matrice d’un *puissance 4* n’est pas carré !

Etape 1 :

$$\begin{pmatrix} a & b & c & d & e \\ f & g & h & i & j \\ k & l & m & n & p \\ q & r & s & t & u \end{pmatrix}$$

$$\times$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} a & b & c & d & e \\ f & g & h & i & j \\ k & l & m & n & p \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

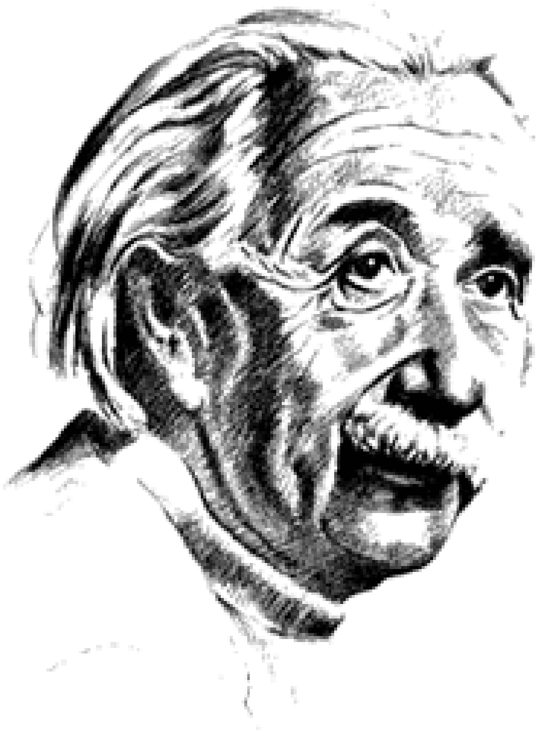
Etape 2 :

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

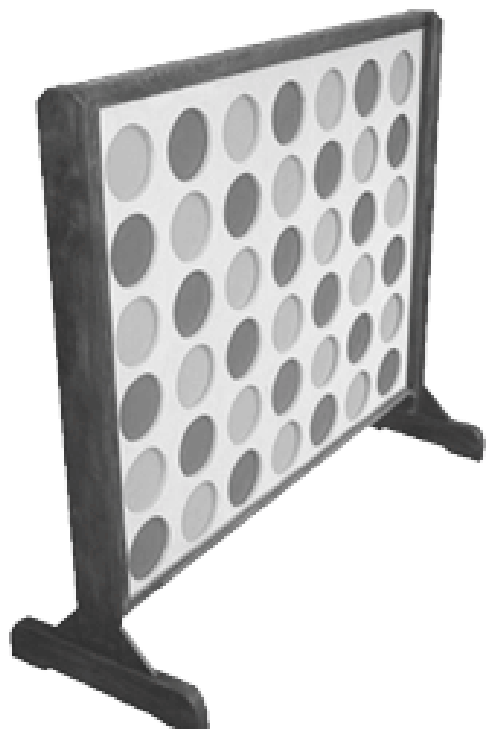
$$\times$$

$$\begin{pmatrix} a & b & c & d & e \\ f & g & h & i & j \\ k & l & m & n & p \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} a & b & c & 0 & 0 \\ f & g & h & 0 & 0 \\ k & l & m & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Les stratégies



Comment faire pour gagner à chaque partie ?



Les stratégies

- Le joueur qui commence, s'il joue au centre et ne fait pas d'erreur, est sûr de gagner.

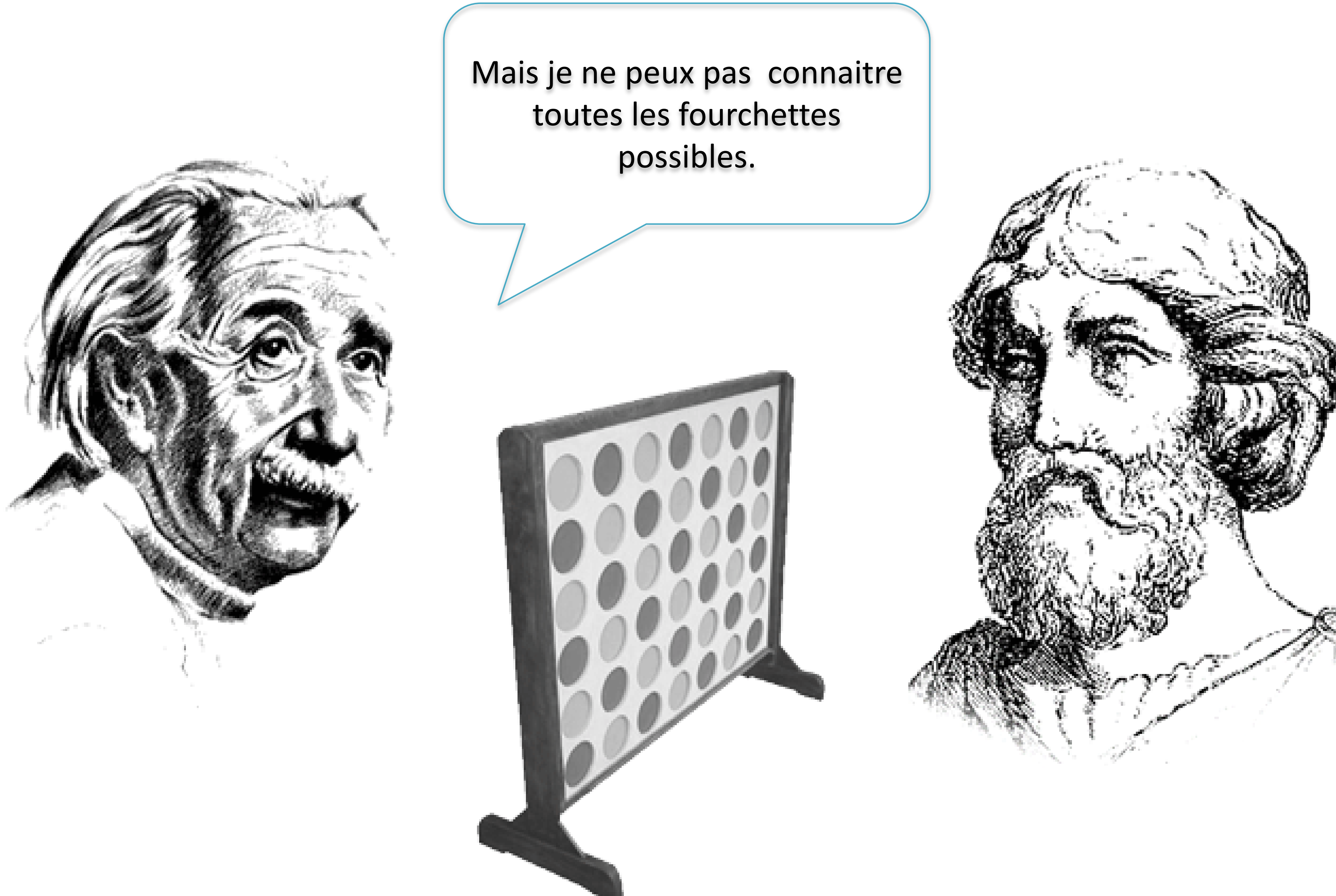
- Il existe des pièges appelés « fourchette » qui oblige un joueur à faire un choix perdant.

Exemple :

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

C'est à Pythagore de jouer.

Mais où qu'il joue il aura perdu.


Mais je ne peux pas connaitre toutes les fourchettes possibles.

Programmation IA

- Donc nous partons du principe qu'Einstein ne se fera pas avoir par un alignement simple. Et que notre IA joue en premier.

- Nous avons programmé* notre IA de façon à ce que celle-ci regarde le jeu, aille chercher dans sa « bibliothèque » la fourchette la plus proche et joue de manière à amener Einstein dans le piège.

** Nous n'avons pas réussi à la programmer car nos connaissances en informatique sont insuffisantes. Il existe des concours d'informatique et mathématique pour la programmation d'une telle IA.*

Programmation IA

Fourchette gagnante
Etape n :

$$\begin{pmatrix} . & . & . & . & . & . & . \\ 0 & 0 & -1 & -1 & 0 & 0 & . \\ 0 & 0 & 1 & 1 & 1 & 0 & . \end{pmatrix}$$

A Einstein (joueur2) de jouer.

Etape n -1 :

$$\begin{pmatrix} . & . & . & . & . & . & . \\ 0 & 0 & -1 & -1 & 0 & 0 & . \\ 0 & 0 & 1 & 1 & 0 & 0 & . \end{pmatrix}$$

A l'ordinateur (joueur1) de jouer.

Etape n-2 :

$$\begin{pmatrix} . & . & . & . & . & . & . \\ 0 & 0 & -1 & 0 & 0 & 0 & . \\ 0 & 0 & 1 & 1 & 0 & 0 & . \end{pmatrix}$$

$$\begin{pmatrix} . & . & . & . & . & . & . \\ 0 & 0 & 0 & -1 & 0 & 0 & . \\ 0 & 0 & 1 & 1 & 0 & 0 & . \end{pmatrix}$$

A Einstein (joueur2) de jouer.

Impossible car nous partons du principe que le premier coup de l'ordinateur est au milieu

$$\begin{pmatrix} . & . & . & . & . & . & . \\ 0 & 0 & 0 & -1 & 0 & 0 & . \\ 0 & 0 & 0 & 1 & 0 & 0 & . \end{pmatrix}$$

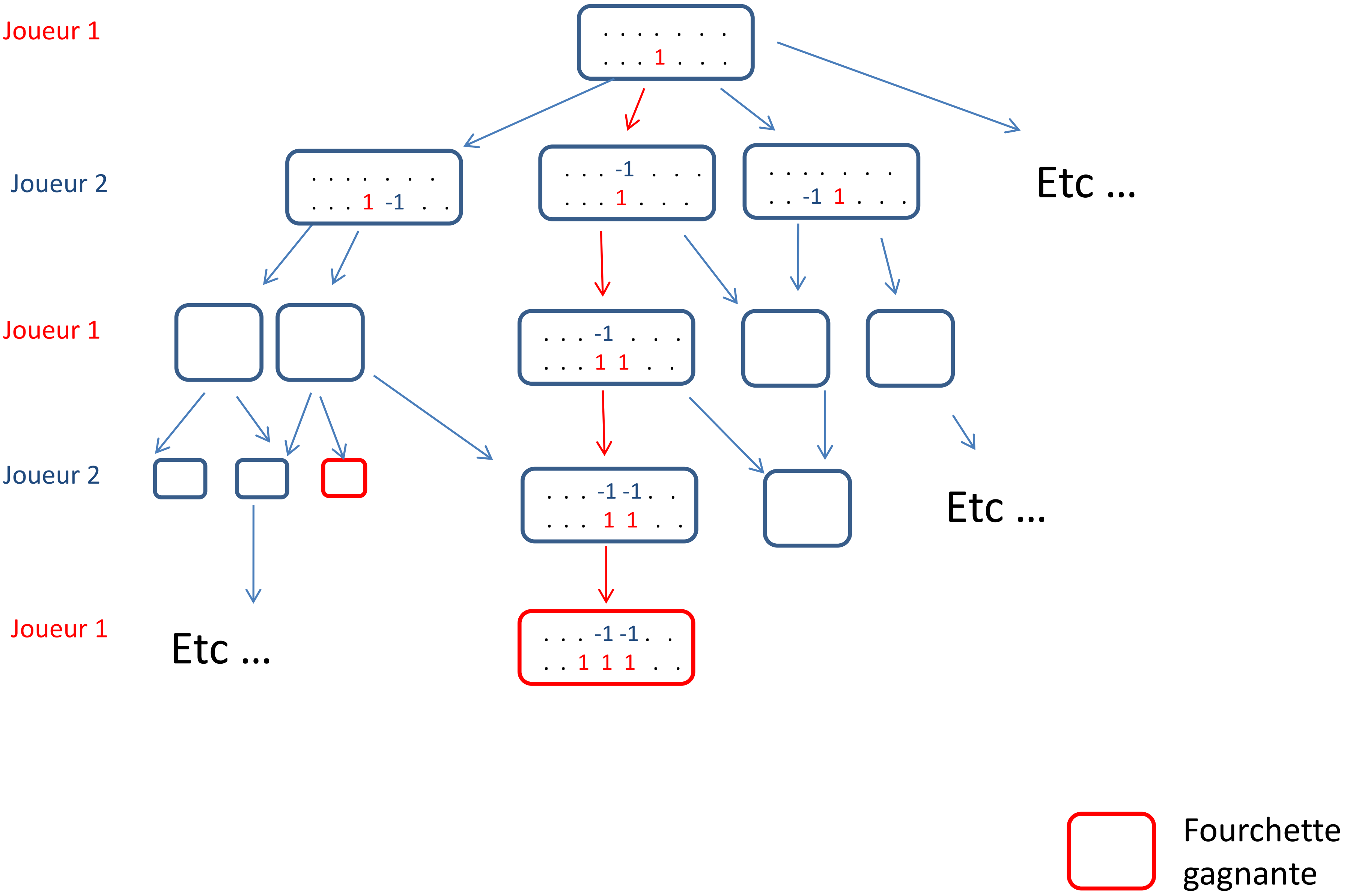
A l'ordinateur (joueur1) de jouer.

Programmation IA

- Après avoir rentré dans la bibliothèque toutes les fourchettes, l'ordinateur va enregistrer chaque coup.

- Suivre un chemin suivant les coups adverses vers la fourchette ayant le moins d'étapes possibles.

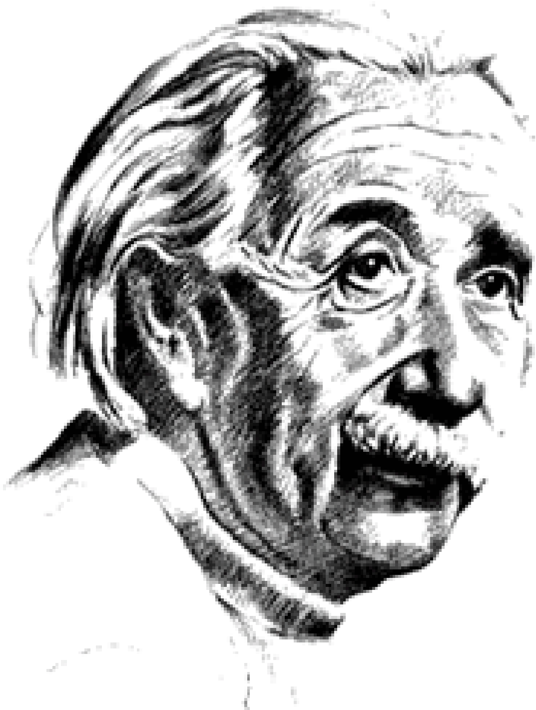
- OU suivre le chemin ayant le plus de fourchette possible.


Joueur 1
Joueur 2
Joueur 1
Joueur 2
Joueur 1
Etc ...
Etc ...
Etc ...
Fourchette gagnante

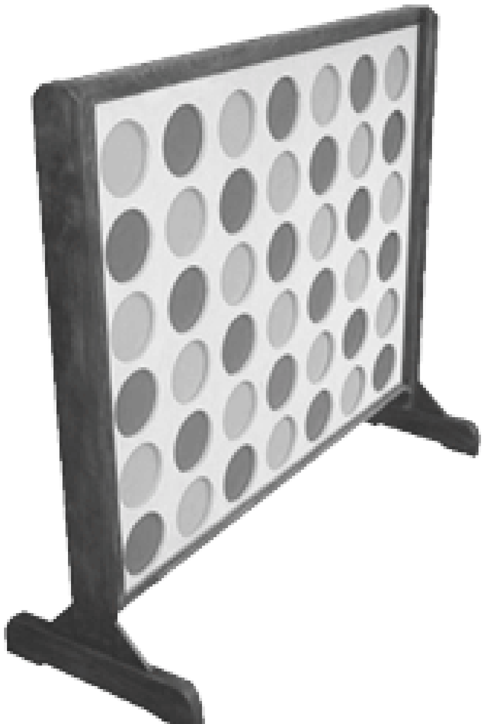

Je suis sûr de gagner en maximum 21 coups !

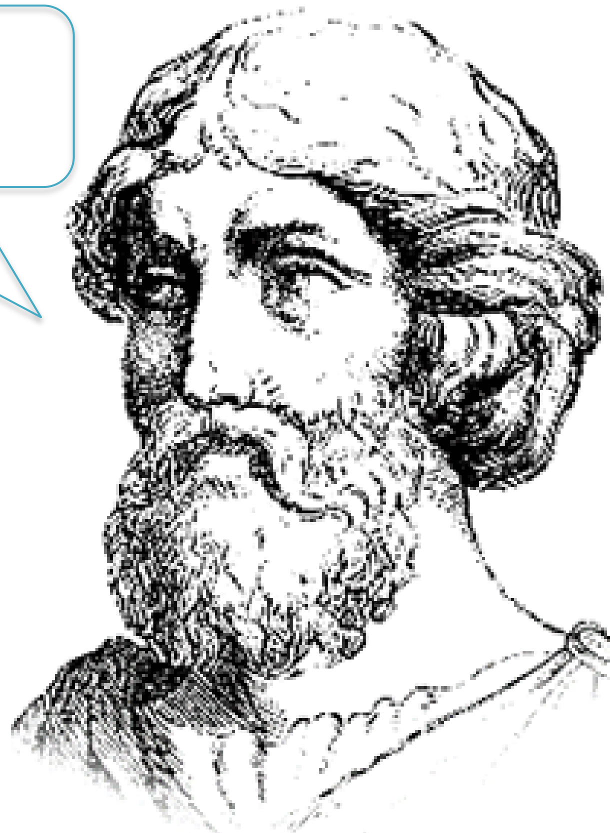

Pourquoi c'est l'ordinateur qui commence ?

Les autres manières d'envisager le *puissance 4*



Mais c'est une grille, c'est forcément une matrice !





Les autres manières d'envisager le *puissance 4*

- « Théorie des poids ». En travaux
- En utilisant la géométrie. Abandonné
- Sous forme de code en base 3. En travaux
- Sous forme matricielle avec des nombres complexes. Abandonné
- Sous forme de suites. Abandonné

Théorie des poids

Définition :

Chaque case dans le *puissance 4* possède une valeur qui définie le poids de celle-ci. Ce poids est déterminé suivant la ligne, la colonne et les règles suivantes.

Les règles :

- La présence d'un jeton adverse augmente le poids des 8 cases adjacentes de 1.

- Si plus de 2 jetons adverses sont allignés, et à une distance minimum de 2 cases, toutes les case sur cette même ligne gagnent +3 à leurs poids.

- Si 2 jetons de la même couleur sont alignés alors les extrêmités de la ligne, colonne ou diagonale gagnent +3.

Exemple :

2	3	4	5	4	3	2
3	4	5	6	5	4	3
4	5	6	7	6	5	4
4	5	6	7	6	5	4
3	4	5	6	5	4	3
2	3	4	5	4	3	2

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

2	3	4	5	4	3	2
3	4	5	6	5	4	3
7	5,5	6	7,5	6	5	4
4	-	9	-	6	5	4
5,5	-	-	-	5	4	3
-	-	-	-	6	3	2

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & -1 & 0 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Sous forme de base 3

Colonnes numérotées à l'envers…

	6	5	4	3	2	1	0
1	729	243	81	27	9	3	1
2	1458	486	162	54	18	6	2

Qu'est-ce que ceci ?

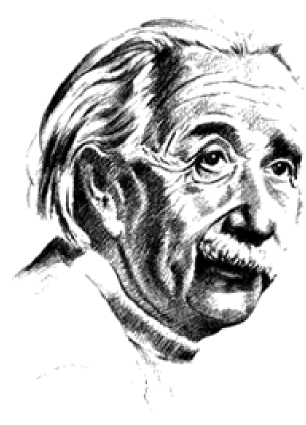
0 0 0 0 0 0 0 : 0
0 0 0 0 0 0 1 : 1
0 0 0 0 0 0 2 : 2
0 0 0 0 0 1 0 : 3
0 0 0 0 0 1 1 : 4
0 0 0 0 0 1 2 : 5
0 0 0 0 0 2 0 : 6
0 0 0 0 0 2 1 : 7
…
0 0 0 1 1 1 1 : 40
…
etc

- On commence par représenter la liste de toutes les combinaisons des lignes (qui sera la même que les colonnes et les diagonales).
- Puis on regarde les coups gagnants.

0 0 0 1 1 1 1 : 40
0 0 0 2 2 2 2 : 80
0 0 1 1 1 1 1 : 121
0 0 2 1 1 1 1 : 202
0 0 1 2 2 2 2 : 161
…
1 1 1 1 1 1 1 : 1093
…
2 2 2 2 2 2 2 : 2186

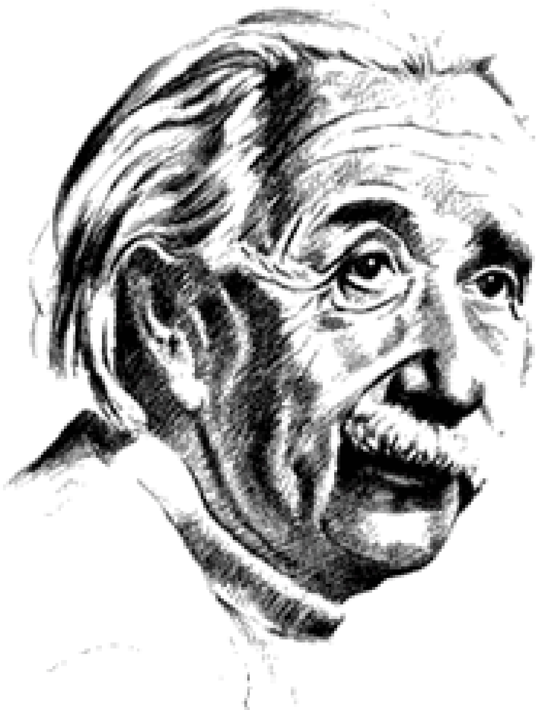
- On commence par représenter la liste de toutes les combinaisons des lignes (qui sera la même que les colonnes et les diagonales).
- Puis on regarde les coups gagnants.
- On récupère la liste des coups combinaisons gagnantes.
- On obtient une suite :

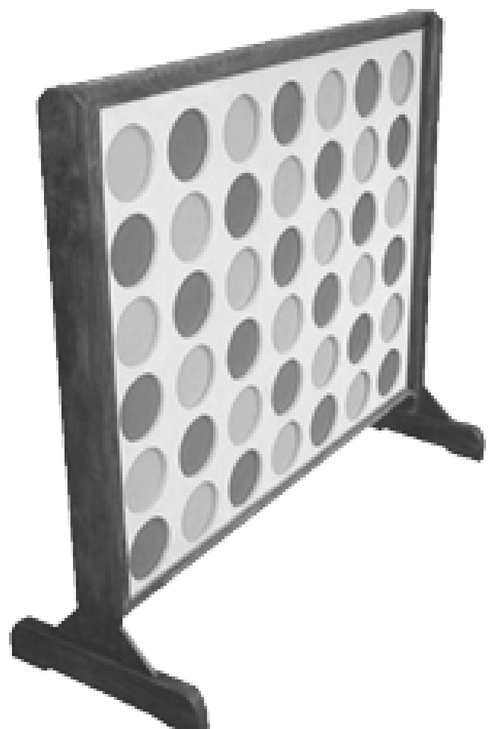
  {40; 80; 121; … ; 2186}

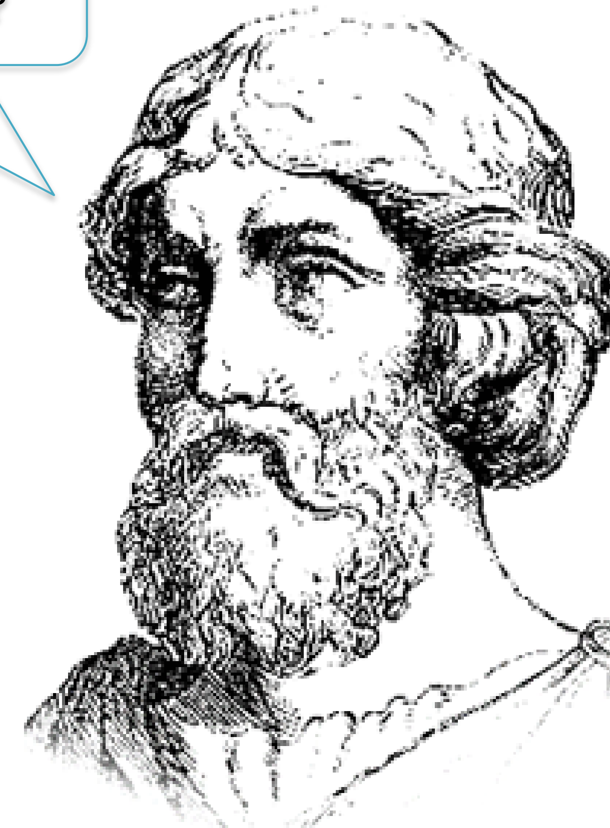


Et qu’est-ce qu’on va faire de cette suite ?

Et après ?

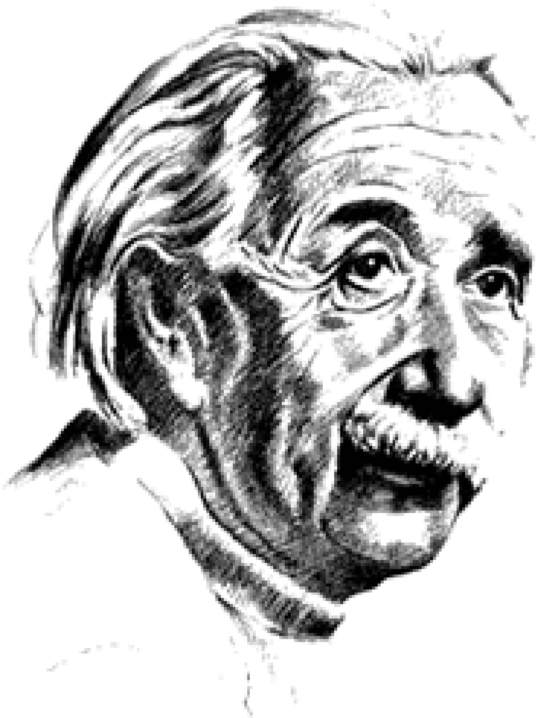


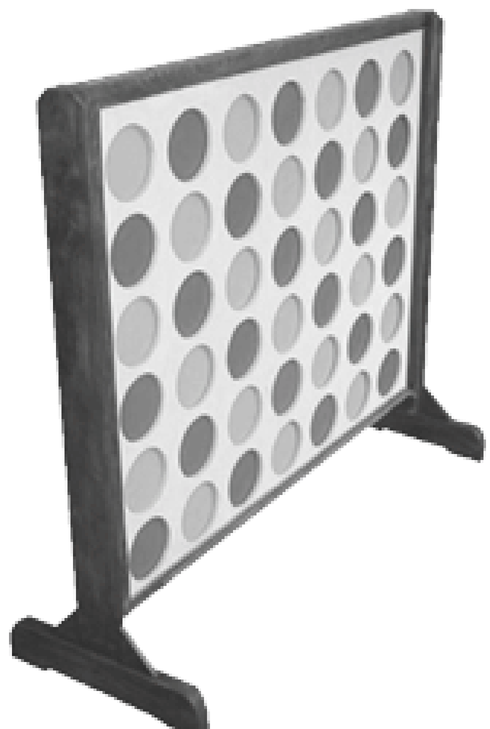




Et après ?

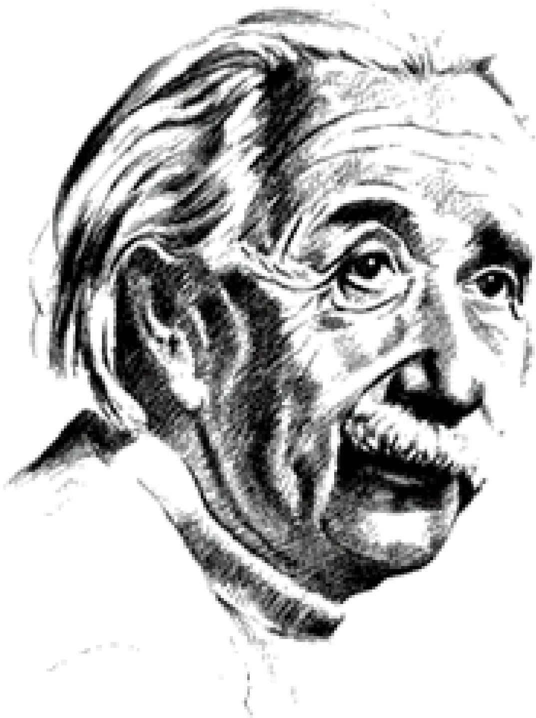
Ou même en 3 dimensions ?

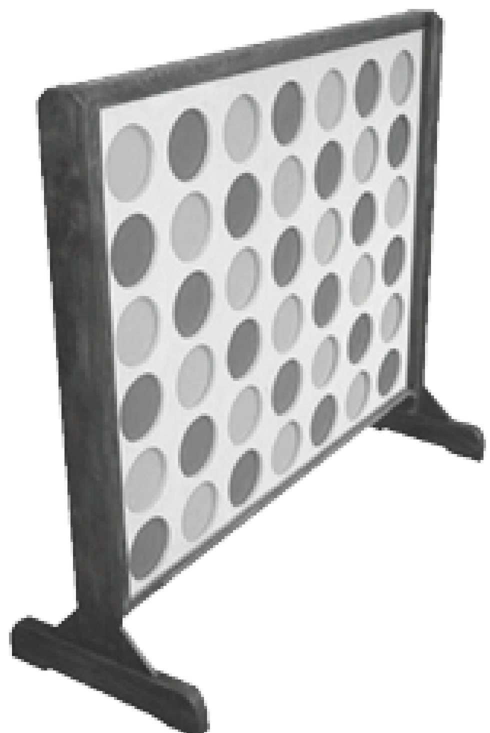


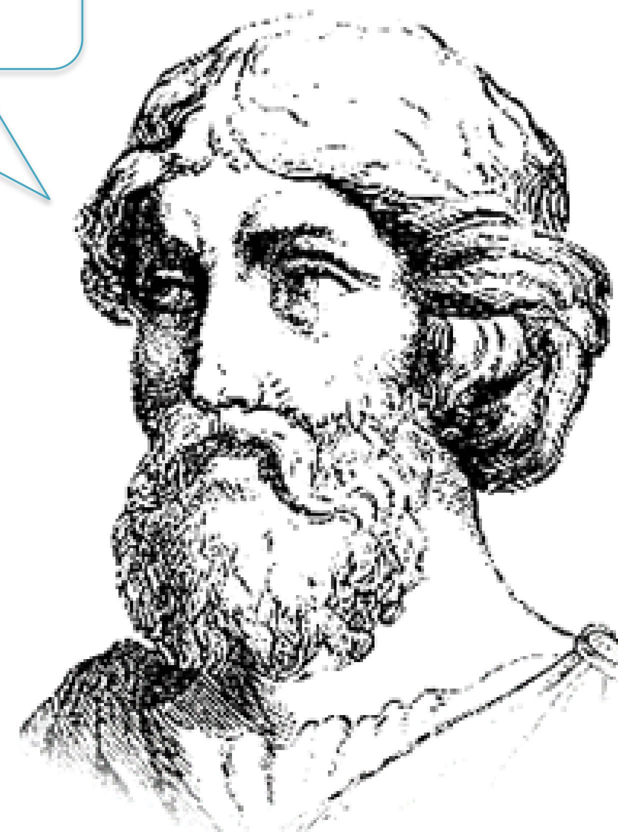


Et après ?

Ou alors avec N joueurs ?







Et après ?

Tant qu’on y est pourquoi pas avec N joueurs et N dimensions !!!