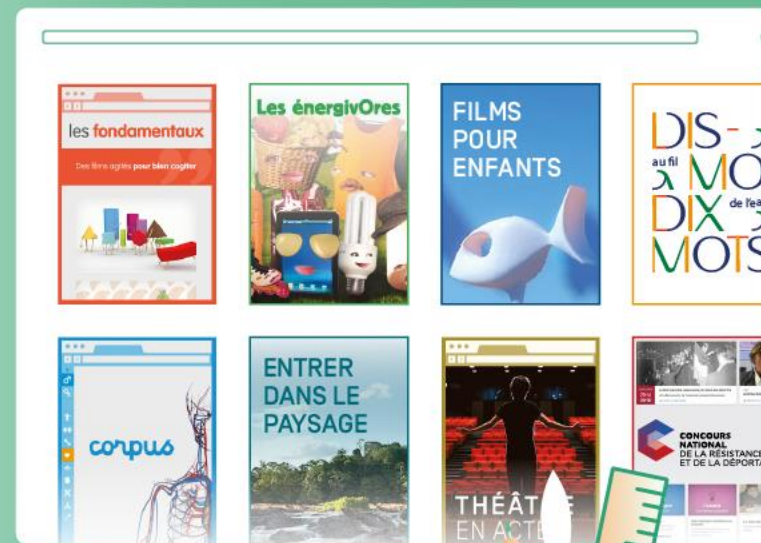


WEBINAIRE

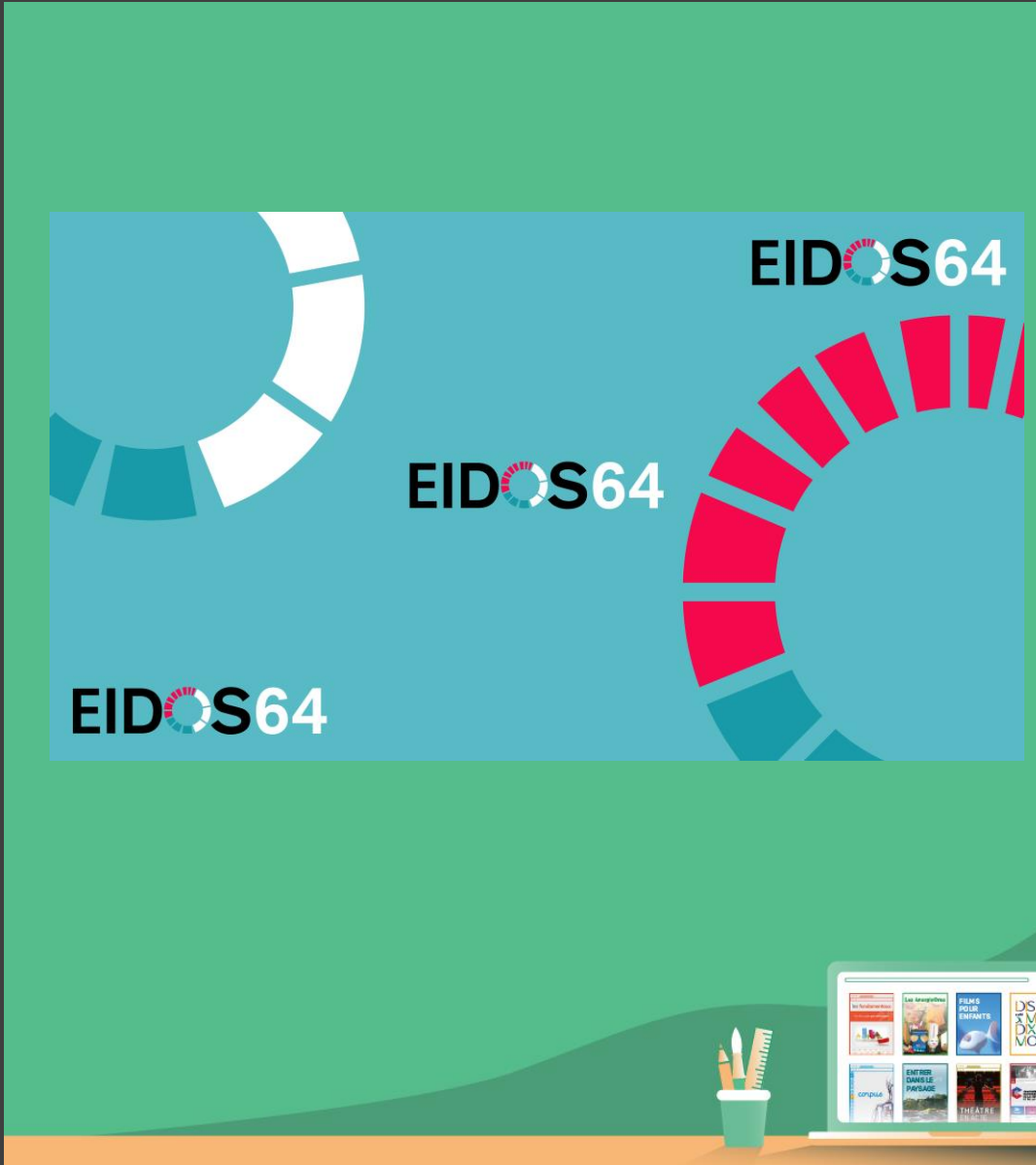


Jeu de Nim et IA
Comprendre l'intelligence
artificielle par l'informatique
débranchée.

Frédéric DUPUY et Emmanuel PAGE

Déroulé

- Qu'est-ce que l'IA ?
- Les 4 concepts informatiques
- Le jeu de NIM (informatique débranchée)
- La machine qui apprend toute seule

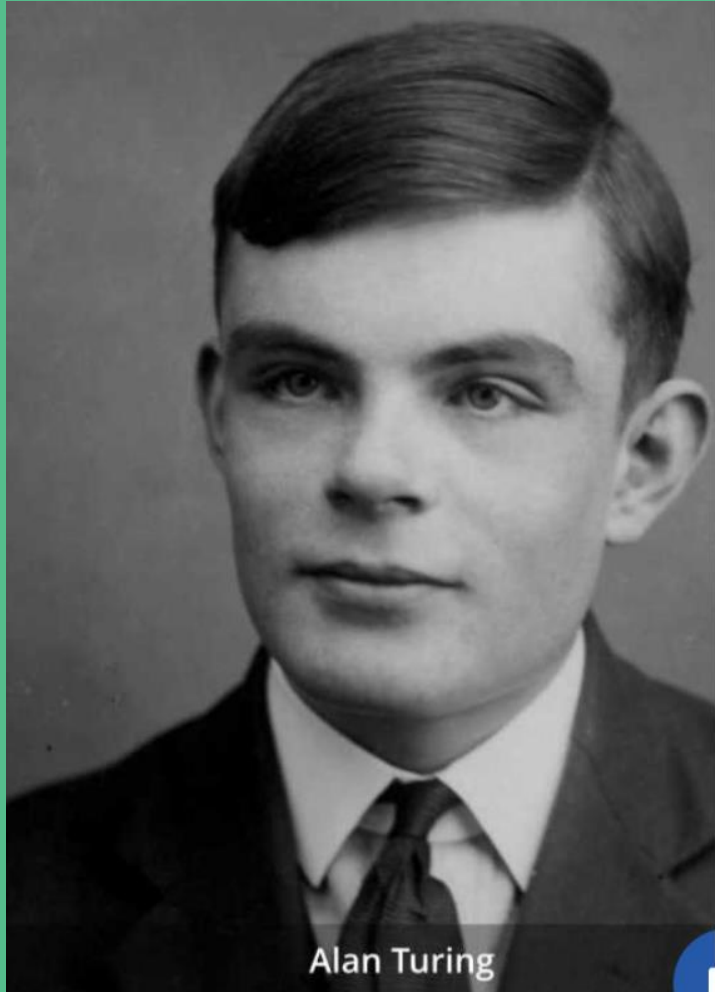


SI JE VOUS PARLE
D'INTELLIGENCE
ARTIFICIELLE
VOUS PENSEZ À ?

À VOUS CHEF !



Qu'est-ce que l'intelligence artificielle ?



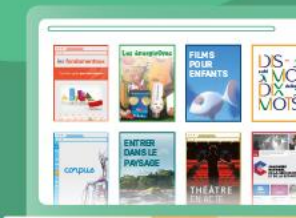
Alan Turing



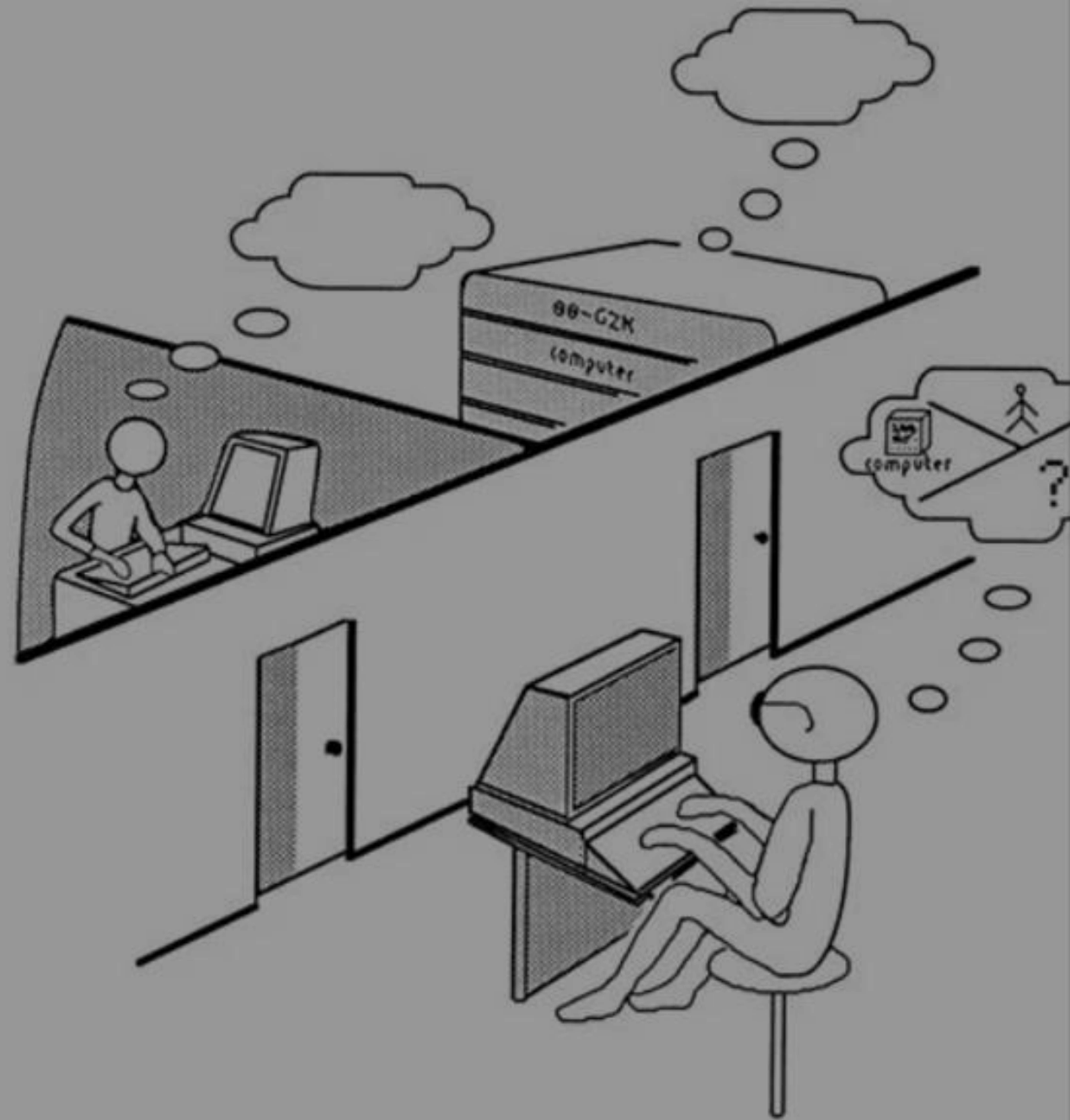
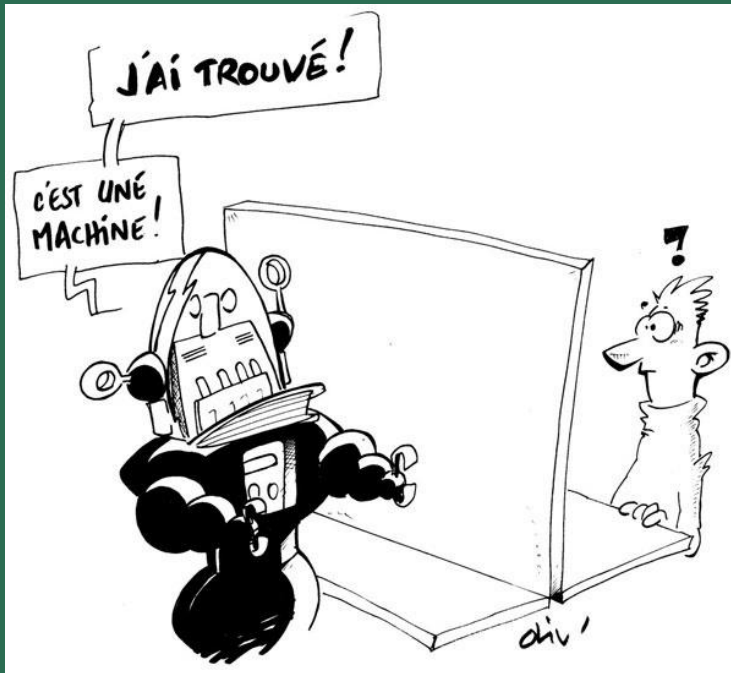
John McCarthy



Marvin Minsky



Test de Turing



Qu'est-ce que l'IA?

Une définition possible :

« Un système **autonome** et **adaptatif** capable de comprendre, prévoir et prescrire sur la base de l'analyse de grandes quantités de données. »

En d'autres termes, l'IA est un domaine scientifique qui utilise des programmes informatiques visant à effectuer, au moins aussi bien que des humains, des tâches et résoudre des problèmes nécessitant un certain niveau d'intelligence.

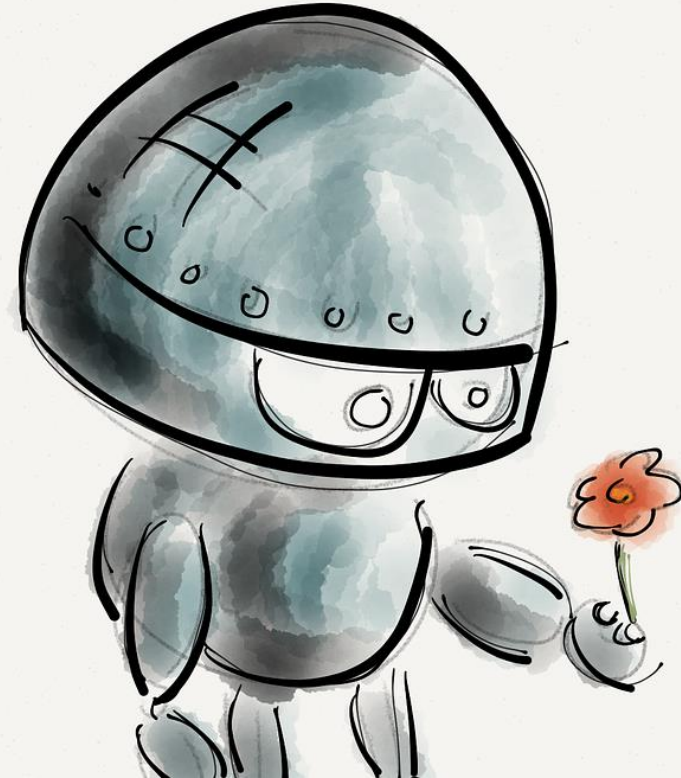




IA Faible

Une technologie basée sur une tâche précise

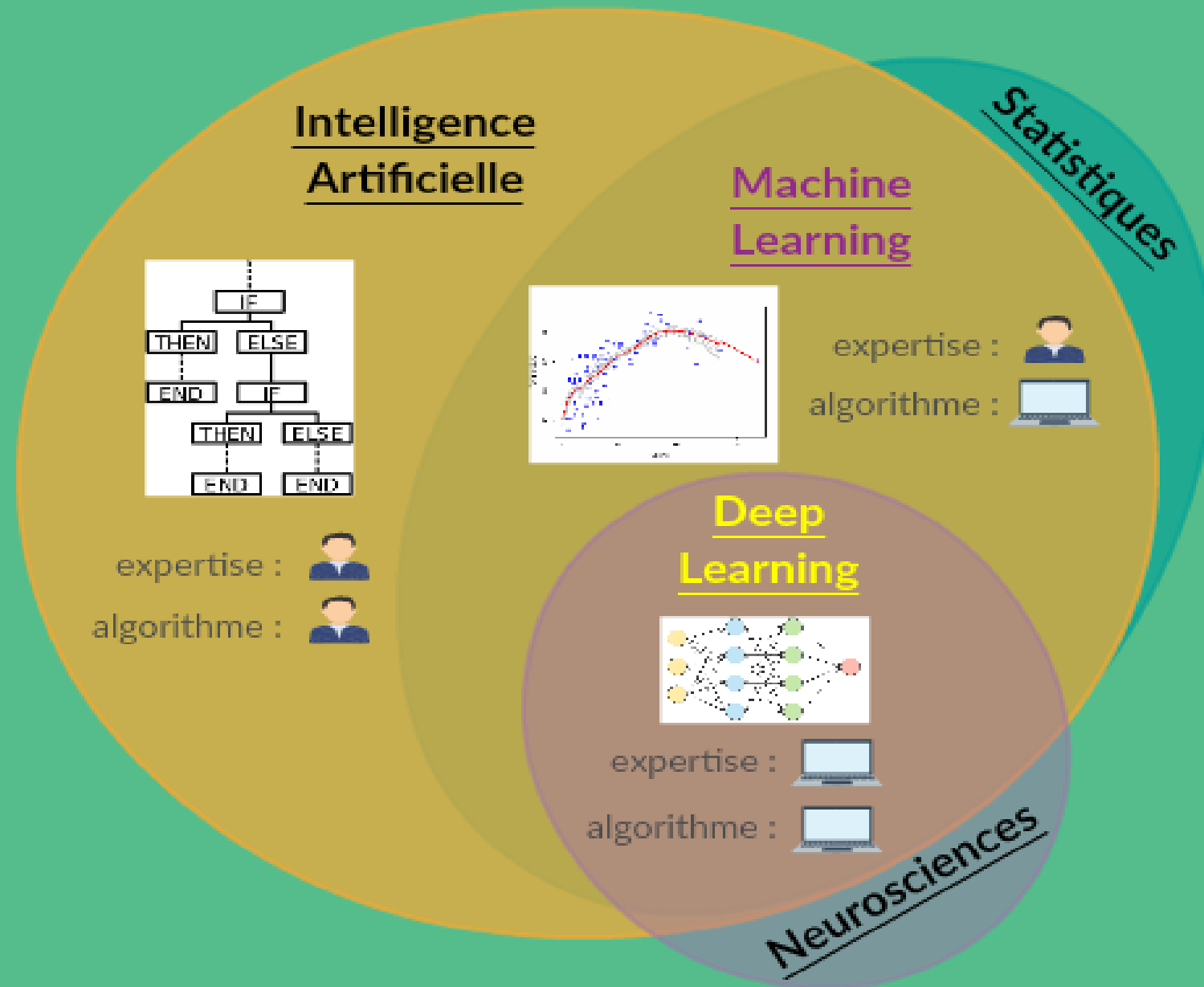




IA FORTE

Une technologie basée sur une imitation du comportement humain complexe, émotionnel et conscient





[Source : Data Science](#)

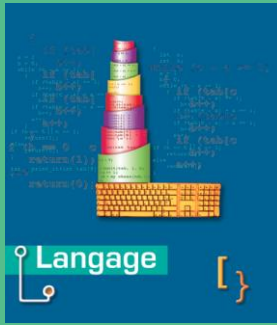
Différence entre Intelligence Artificielle, Machine Learning et Deep Learning - Pensée Artificielle



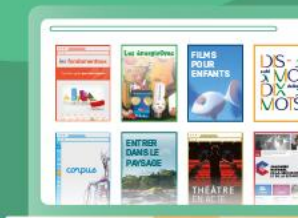


Les 4 concepts de l'informatique

Le concept de « Langage »



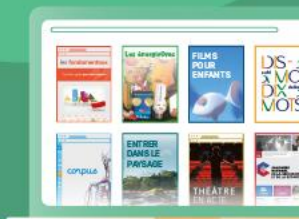
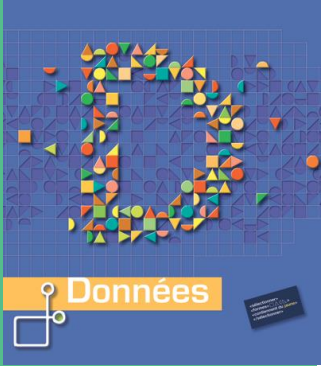
Grace Hopper
1906-1992



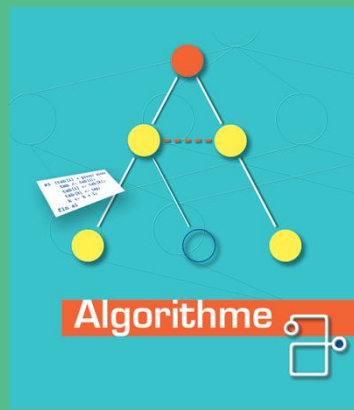
Le concept de « Données »



Alan Turing
1912-1952



Le concept d'« Algorithme »



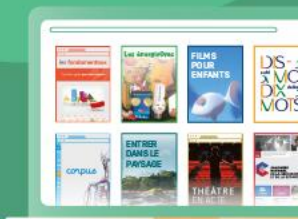
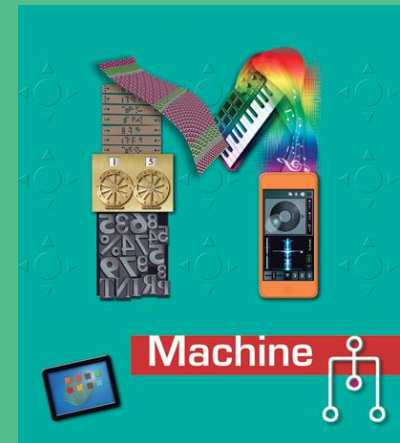
Al-Khwârizmî
IX ème siècle



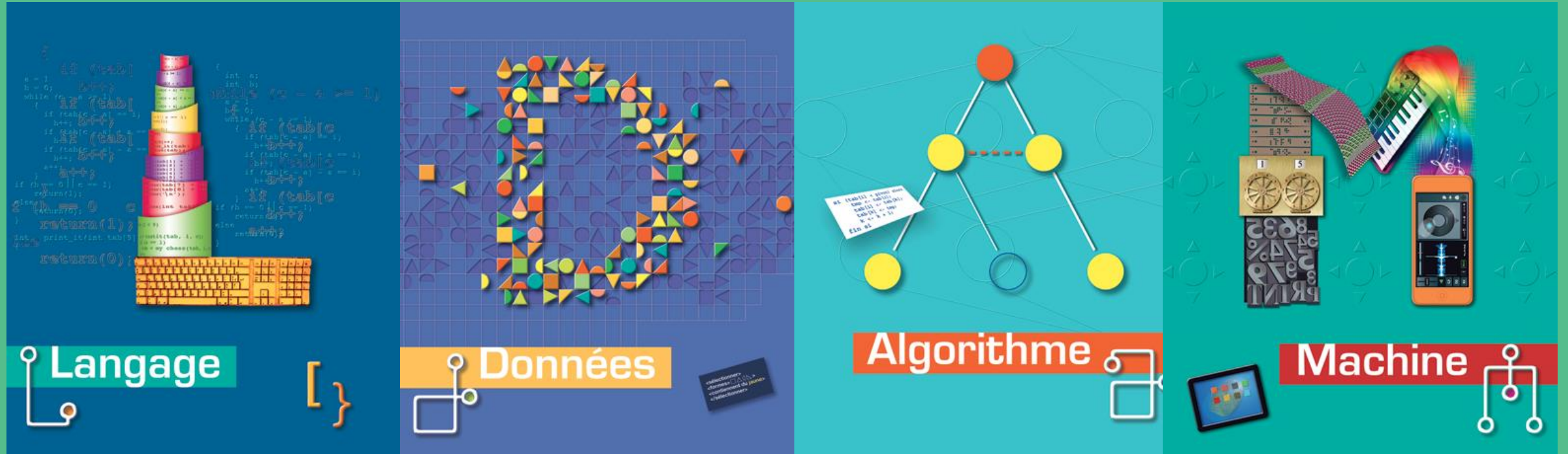
Le concept de « Machine »



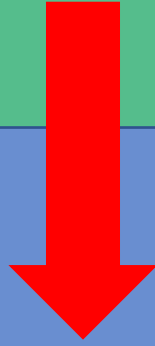
Ada
Lovelace
1815-1852



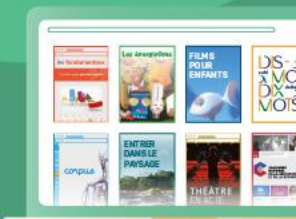
Les sciences du numérique, c'est à la fois :



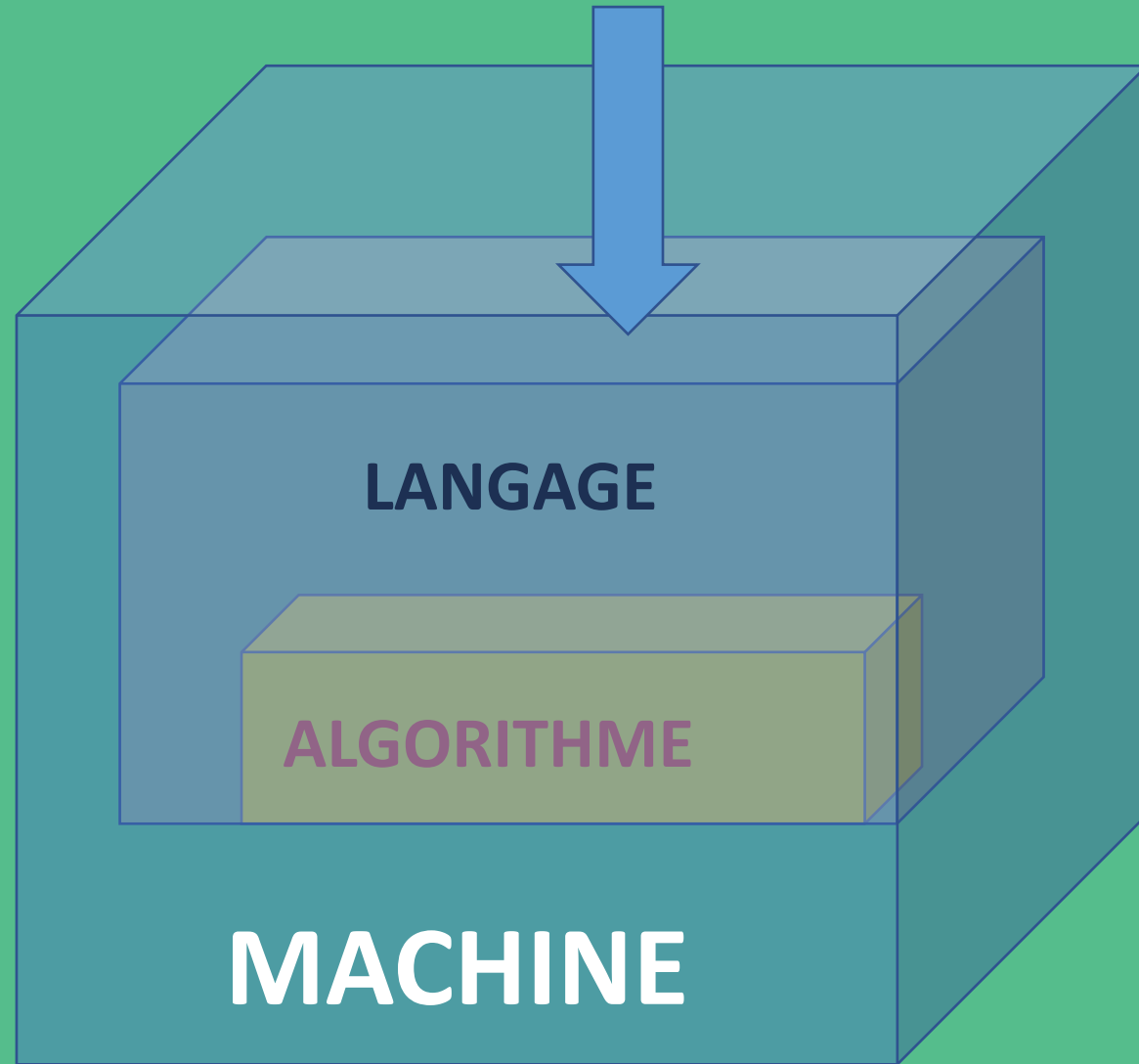
DONNEES/INFORMATIONS

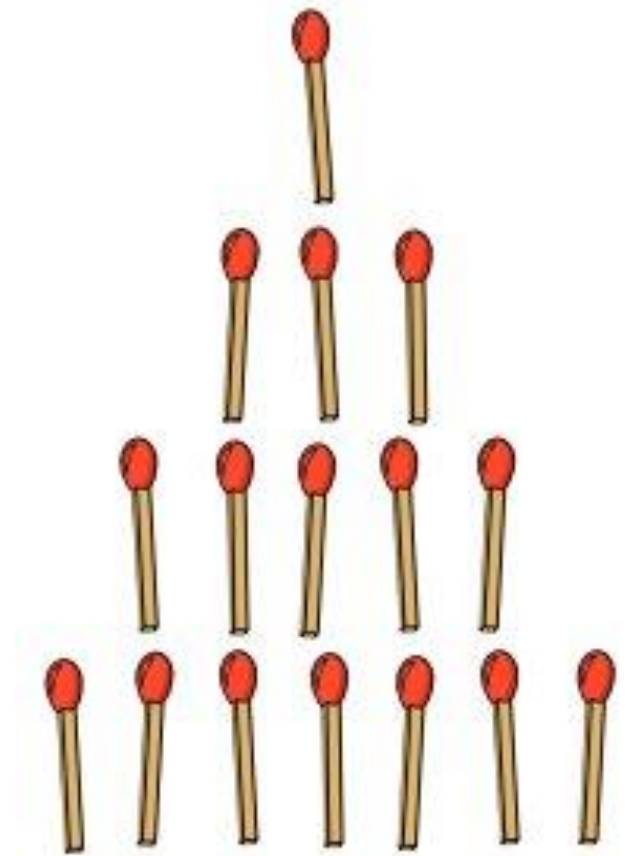


MACHINE

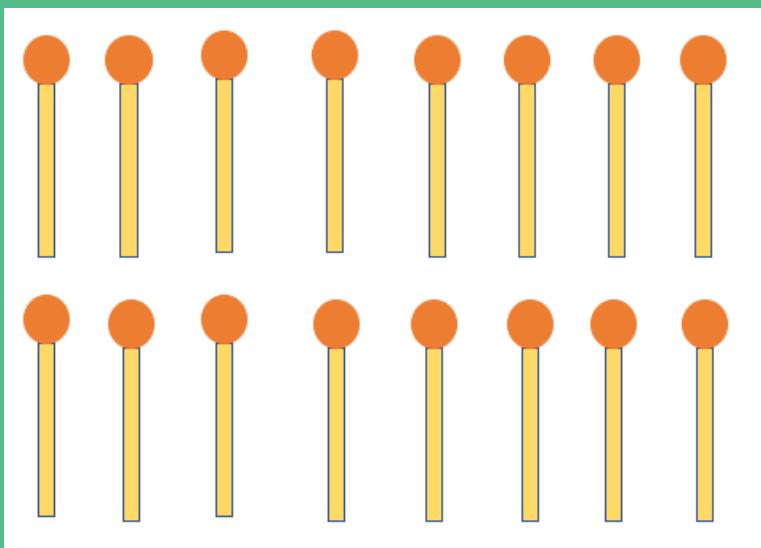


DONNEES/INFORMATIONS





Le jeu de Nim

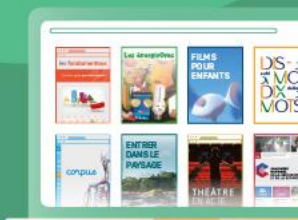


Avec des allumettes, des bouchons, des
jetons ou des nounours à la guimauve ...

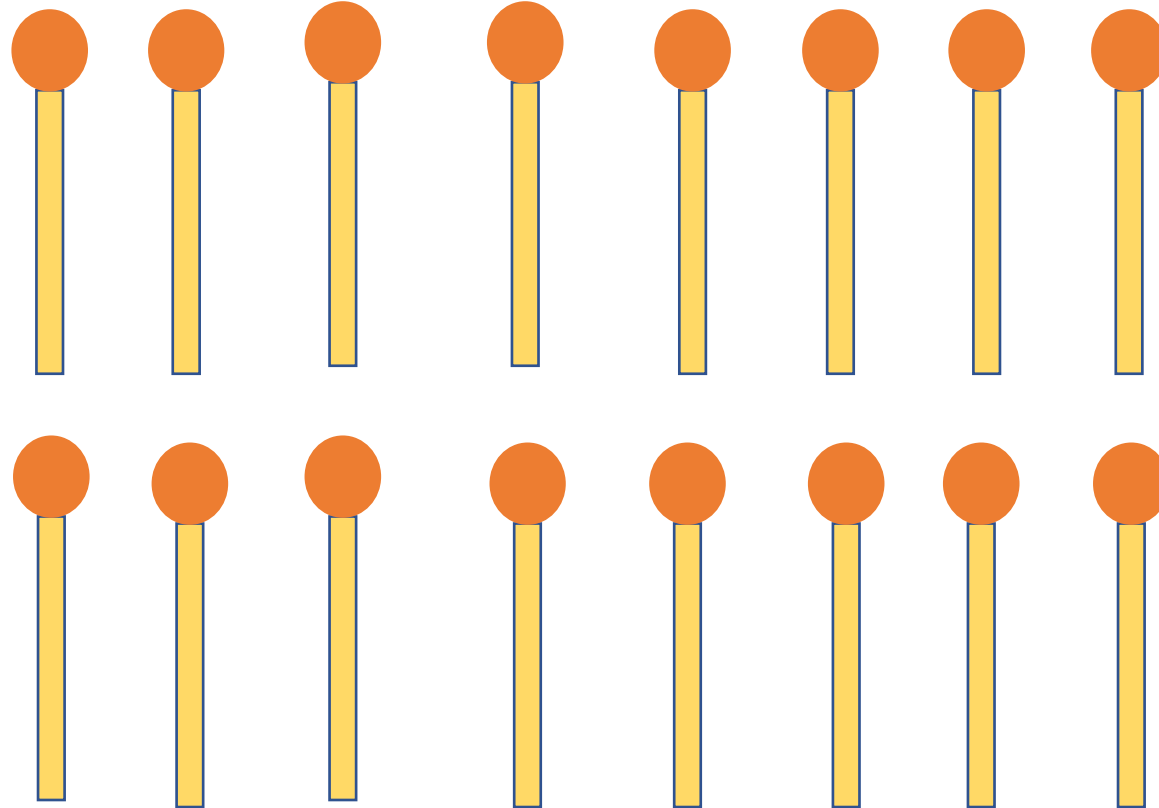


Règles du jeu de Nim

- Deux joueurs (+ 1 observateur) avec 16 objets posés sur une table.
- Le joueur 1 prend un, deux ou trois objets puis c'est au tour du joueur 2 de prendre un, deux ou trois objets.
- Le joueur gagnant est celui qui prend le dernier objet (ou les deux ou trois derniers objets).



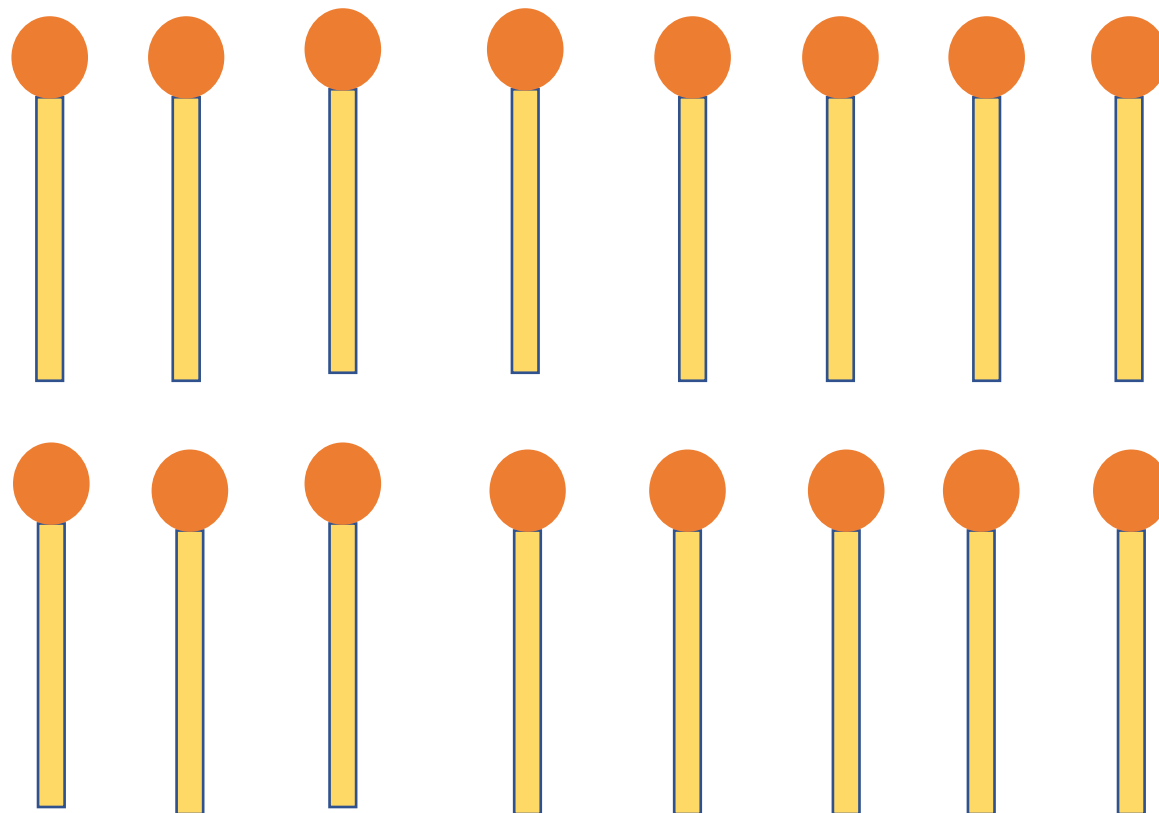
JOUEUR 1



JOUEUR 2

JOUEUR 1

Jouez seul(e) ou à 2 chez vous et essayer de trouver la stratégie pour
gagnez à tous les coups.



JOUEUR 2

FIXER LA REGLE

Pour gagner à chaque fois

Pour gagner, il faut
toujours pouvoir
choisir qui va
commencer

GAGNER AU JEU DE NIM



Celui qui prend la dernière allumette **gagne**

Combien y a-t-il d'allumettes ?

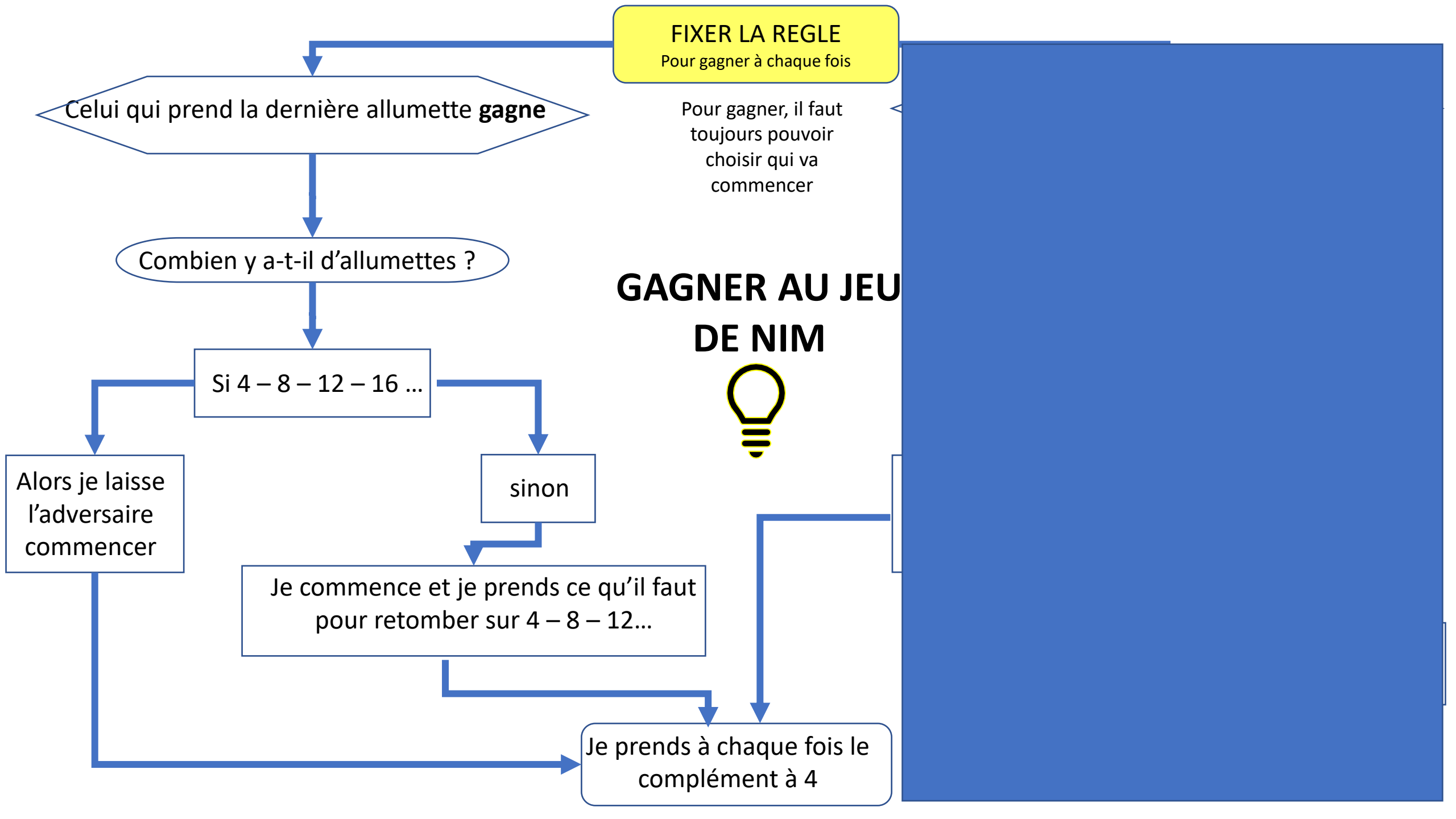
Si 4 – 8 – 12 – 16 ...

sinon

Alors je laisse
l'adversaire
commencer

Je commence et je prends ce qu'il faut
pour retomber sur 4 – 8 – 12...

Je prends à chaque fois le
complément à 4

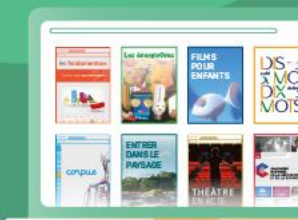


Le Jeu de Nim en classe

Etape 1: Manipuler et jouer

Etape 2 : Verbaliser

Etape 3 : Ecrire l'algorithme de résolution (passer à l'abstraction)



ASTUCE INFANTILE!

si on peut le prendre le dernier objet
est une position perdante.

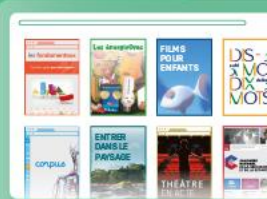
si il prend ④
alors je prends ①
j'ai gagné

si il prend ③
alors je prends ②
j'ai gagné

si il prend ②
alors je prends ③
j'ai gagné

si il prend ①
alors je prends ④
j'ai gagné

Un exemple
d'écriture de
l'algorithme par une
classe de cycle 3
(circonscription
Evian)



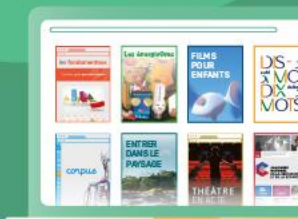
L'adversaire prend
? jetons

Si 1 jeton pris
alors en prendre 3

Si 2 jetons pris
alors en prendre 2

Si 3 jetons pris
alors en prendre 1

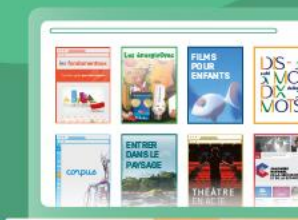
**On commence à écrire le programme avec
4 lignes d'instructions.**



L'adversaire prend
? jetons

Prendre le
complément à 4

On optimise la taille du programme en passant de 3 à 1 instruction.



L'adversaire prend
? jetons

Prendre le
complément à 4

Tant qu'il y a des
jetons sur la table

On introduit la notion de boucle.

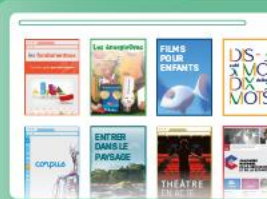


Le Jeu de Nim (un programme)

Le jeu de Nim

Stratégie gagnante "expliquée à l'ordinateur"

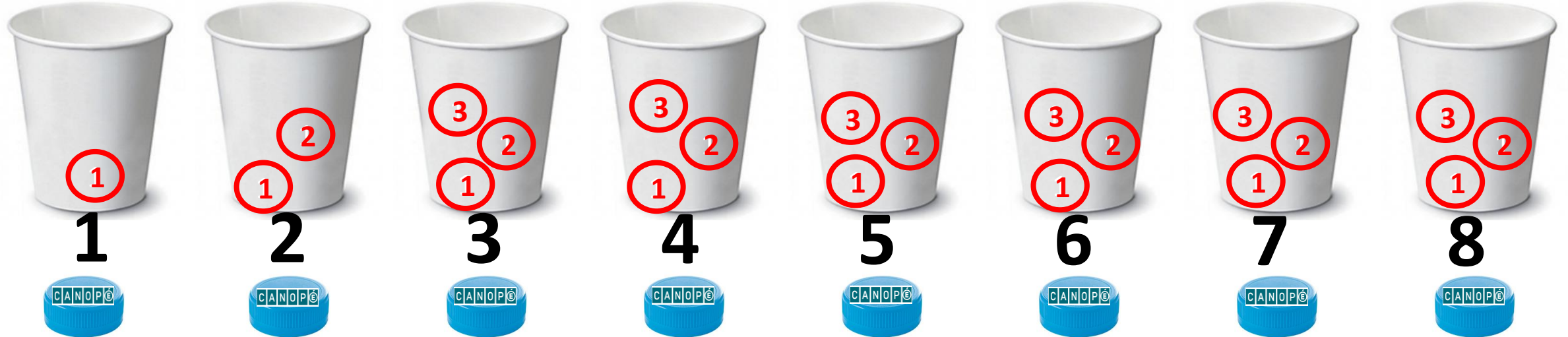
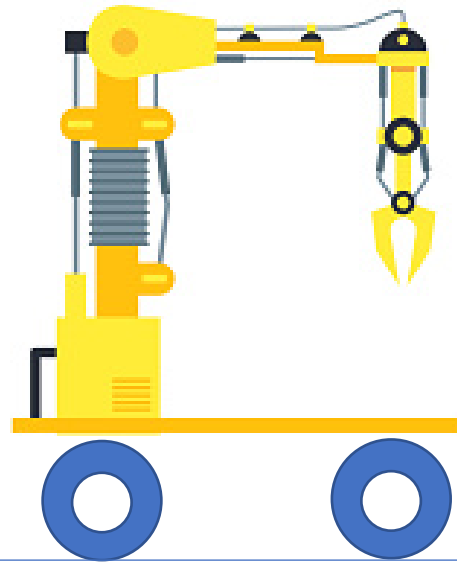
```
import random
def a_mon_tour (nombre d'allumettes)
    reste = nombre_d_allumettes % 4
    # modulo, le reste de la division entière
    if reste == 0:
        # je perds à tous les coups
        prendre (random.randint (1,3))
    else:
        prendre (reste)
```



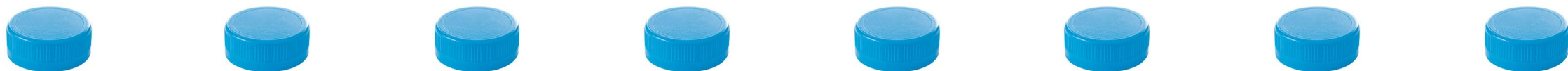
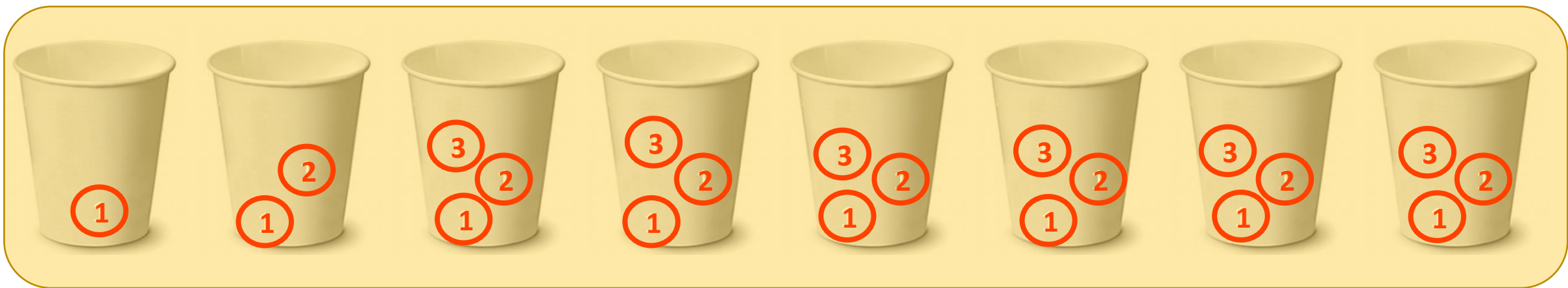


La machine qui apprend toute seule

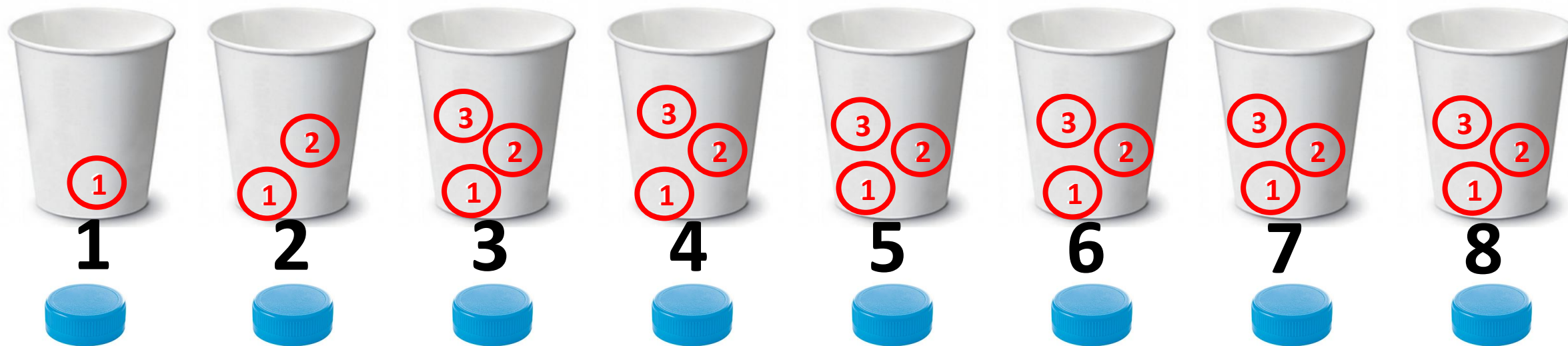
SITUATION DE DEPART



Ces gobelets et ces jetons représentent l'algorithme que va suivre la machine.



TOUR 1 : l'humain prend 1, 2 ou 3 bouchons.

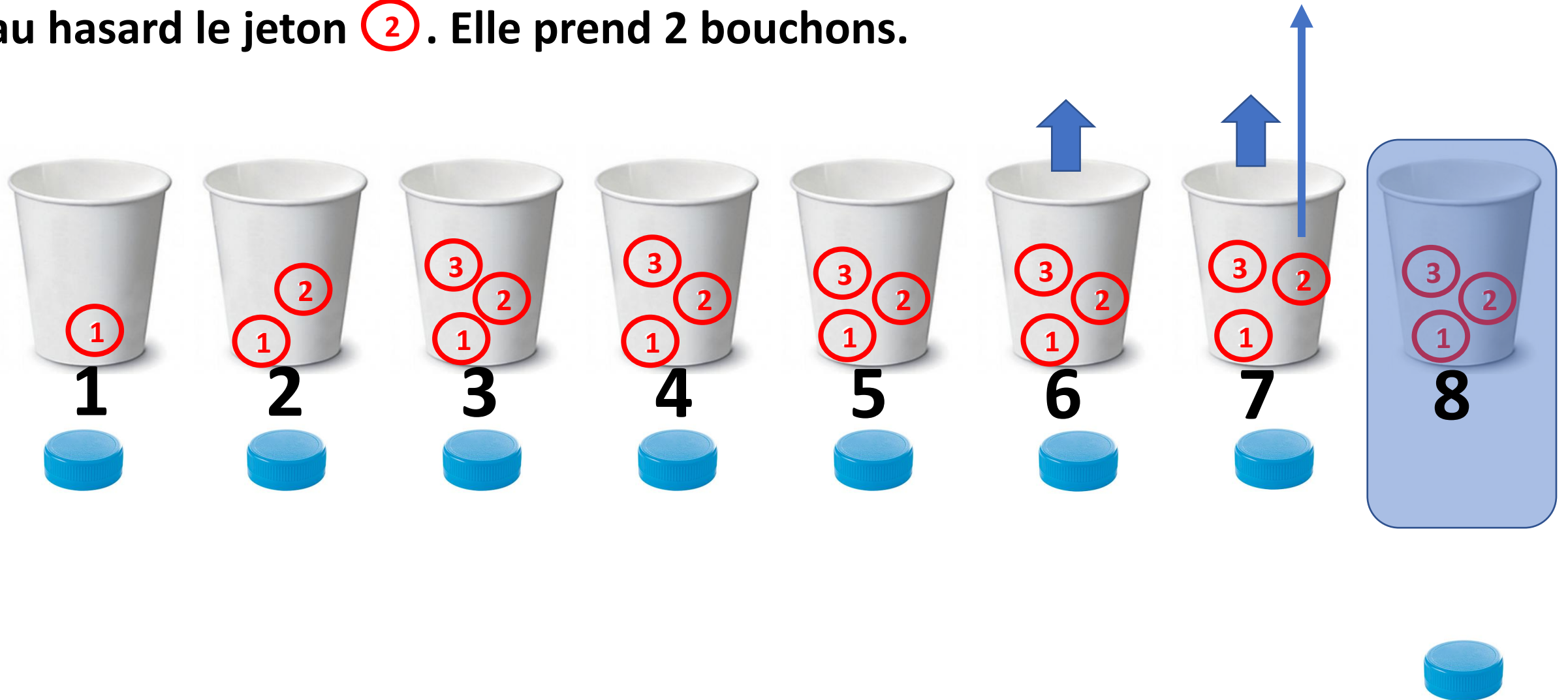


L'humain prend 1 bouchon.

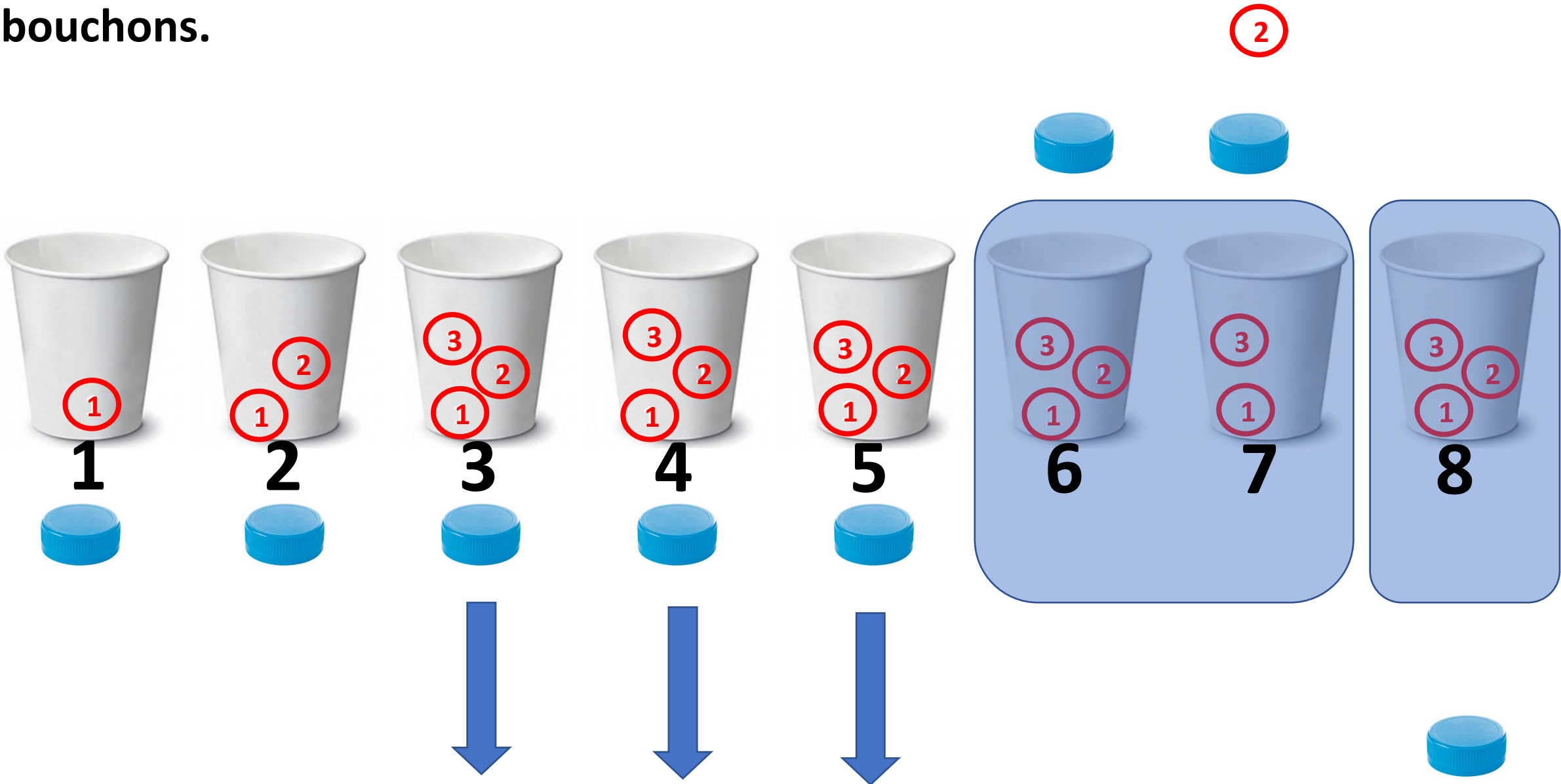


SENS DE JEU

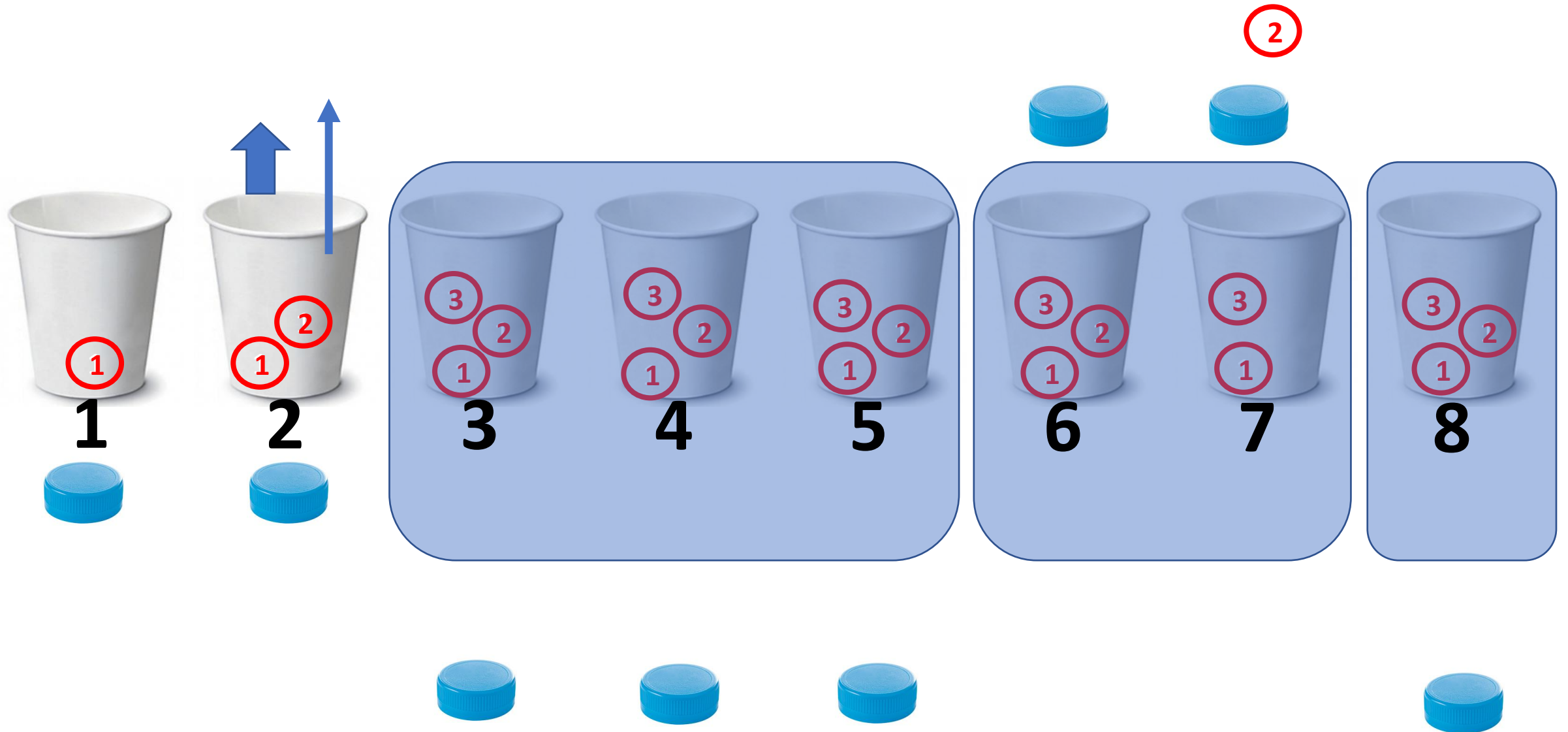
TOUR 1 : la machine joue. Elle voit qu'il reste 7 bouchons, elle se rend au gobelet 7. Elle tire au hasard le jeton 2. Elle prend 2 bouchons.



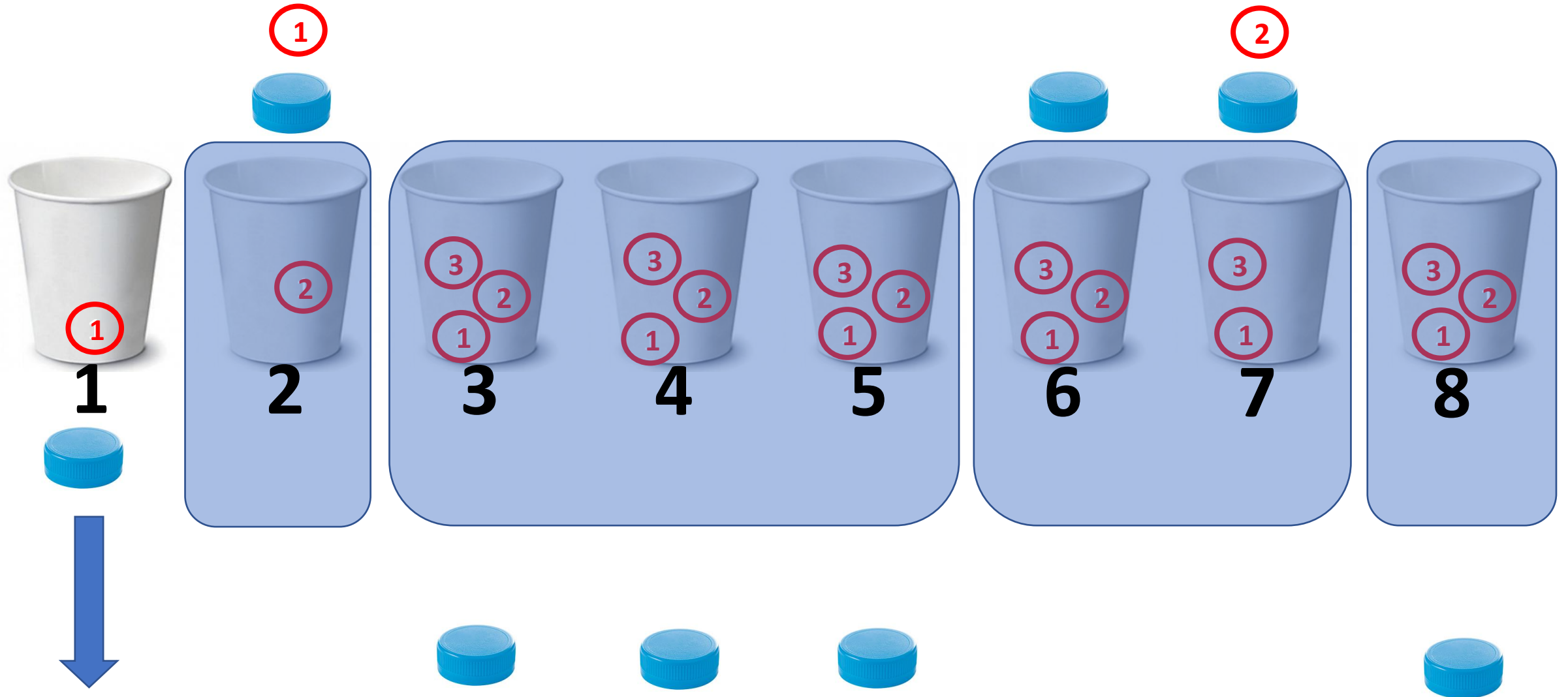
TOUR 2 : l'humain joue. Il prend 3 bouchons.



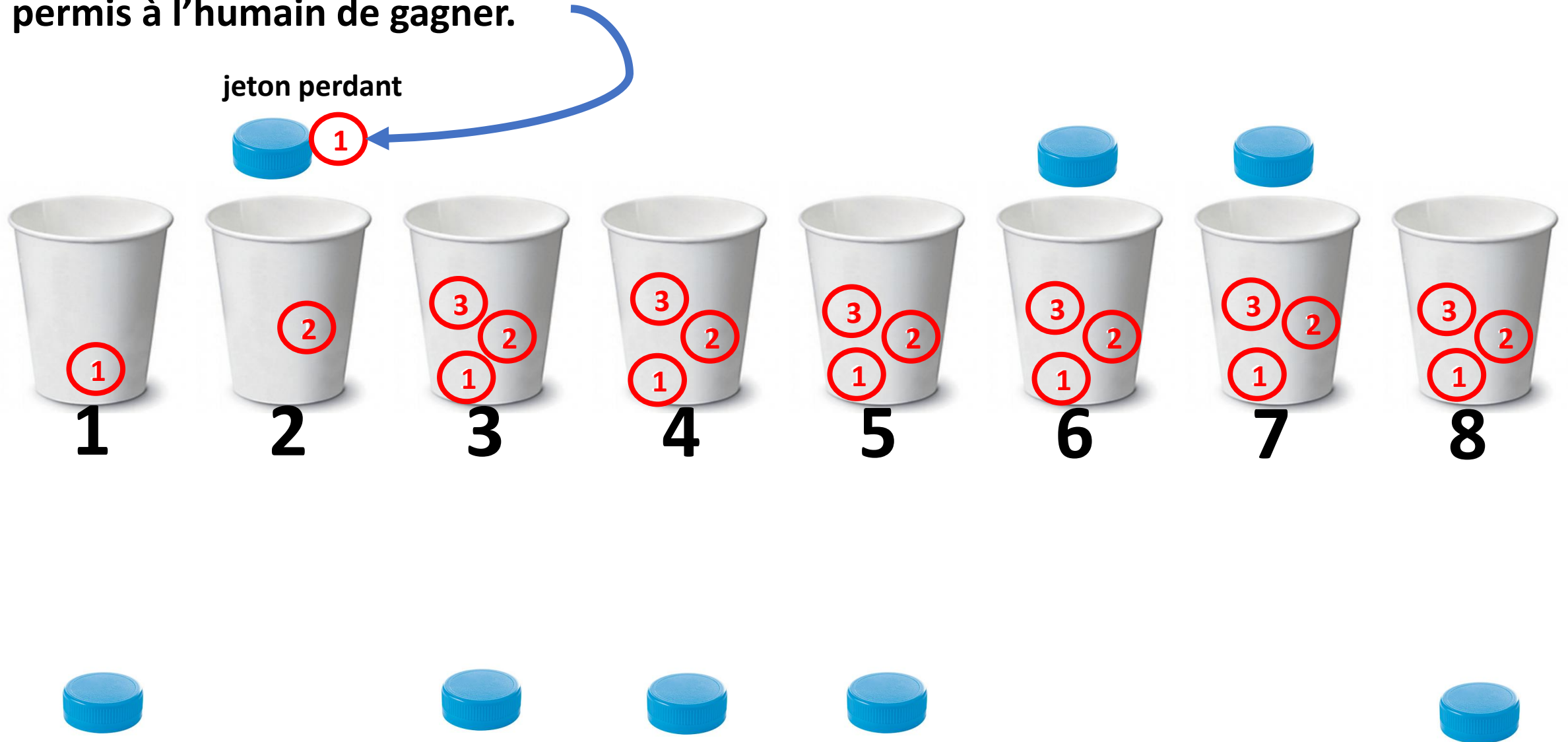
TOUR 2 : la machine joue. Elle tire au hasard le jeton ①. Elle prend 1 bouchon.



TOUR 2 : l'humain joue. Il prend le dernier bouchon, il a gagné. La machine a perdu.

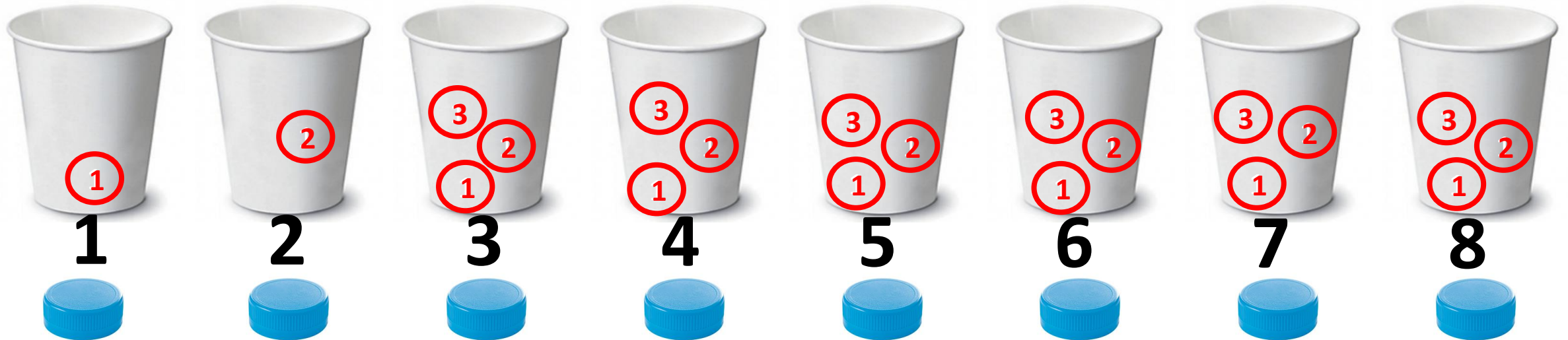


Fin de la 1^{ère} partie : la machine a perdu. Le dernier jeton pris par la machine n'est pas remis dans le gobelet. Il est reconnu comme perdant par la machine, puisque ce jeton a permis à l'humain de gagner.

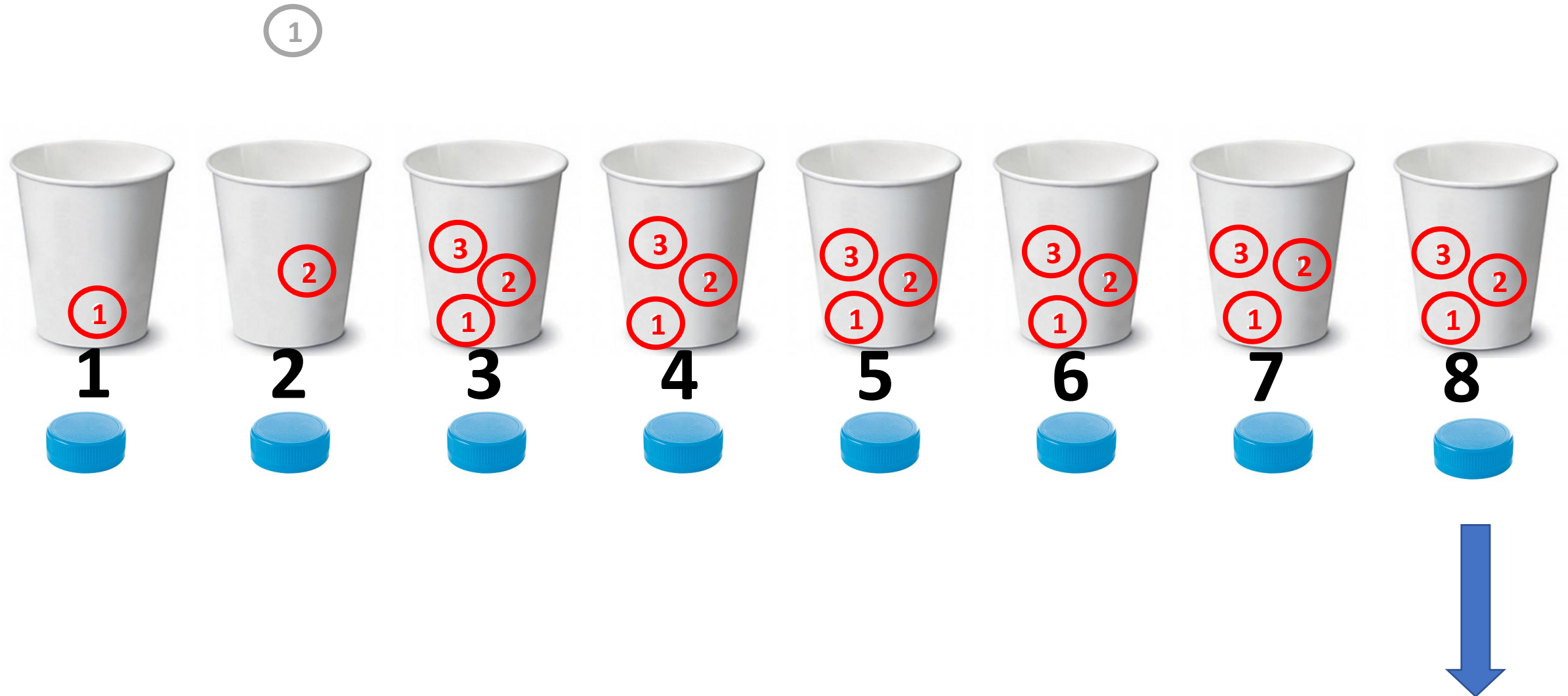


PARTIE 2 : SITUATION DE DEPART

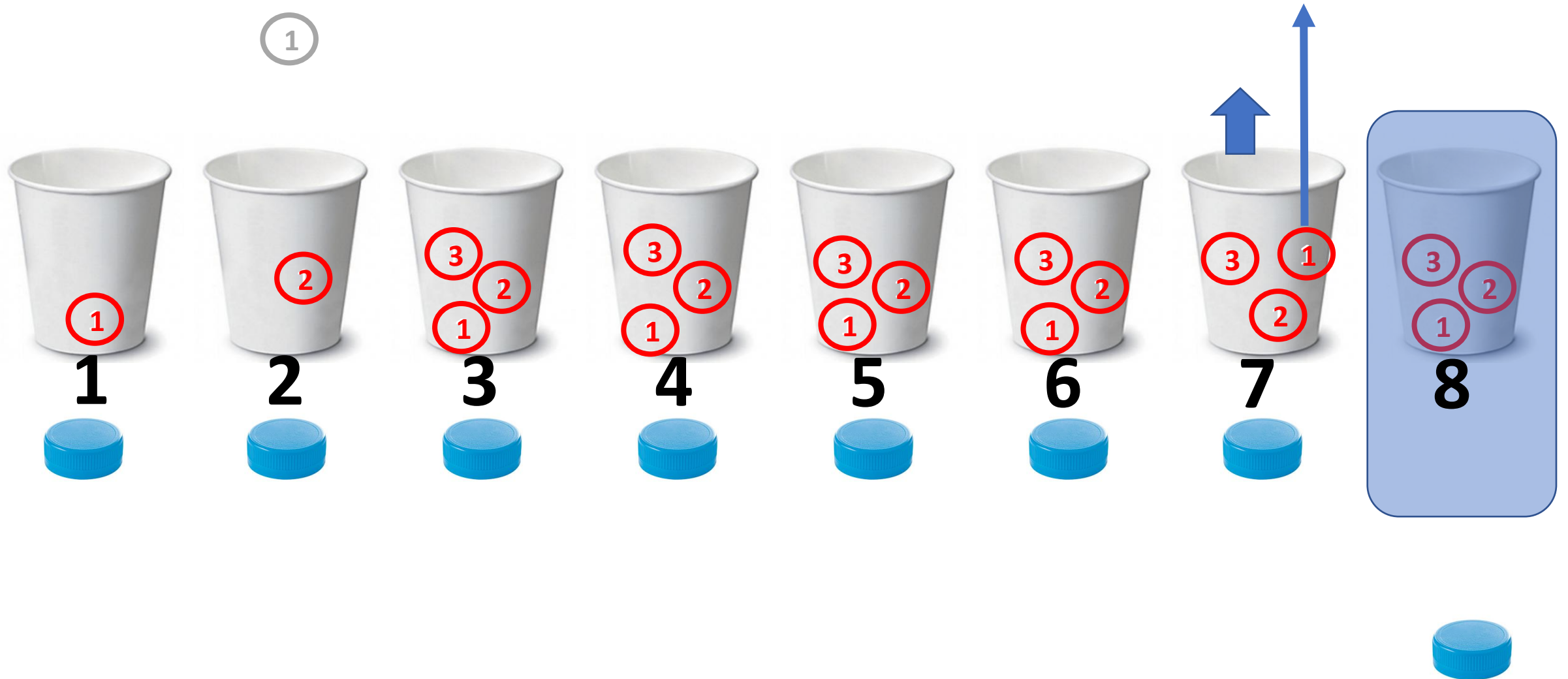
① jeton sorti du jeu



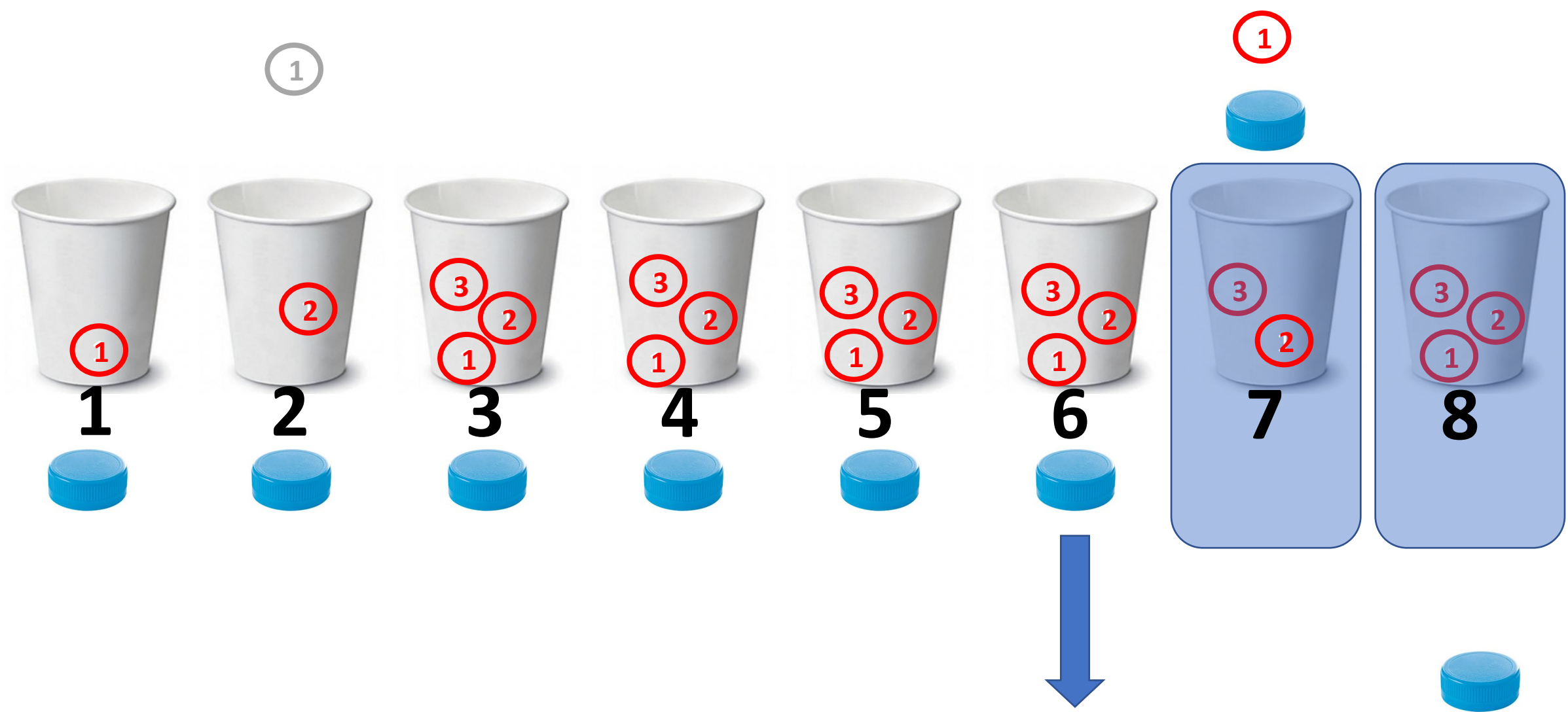
TOUR 1 : l'humain joue et prend 1 bouchon.



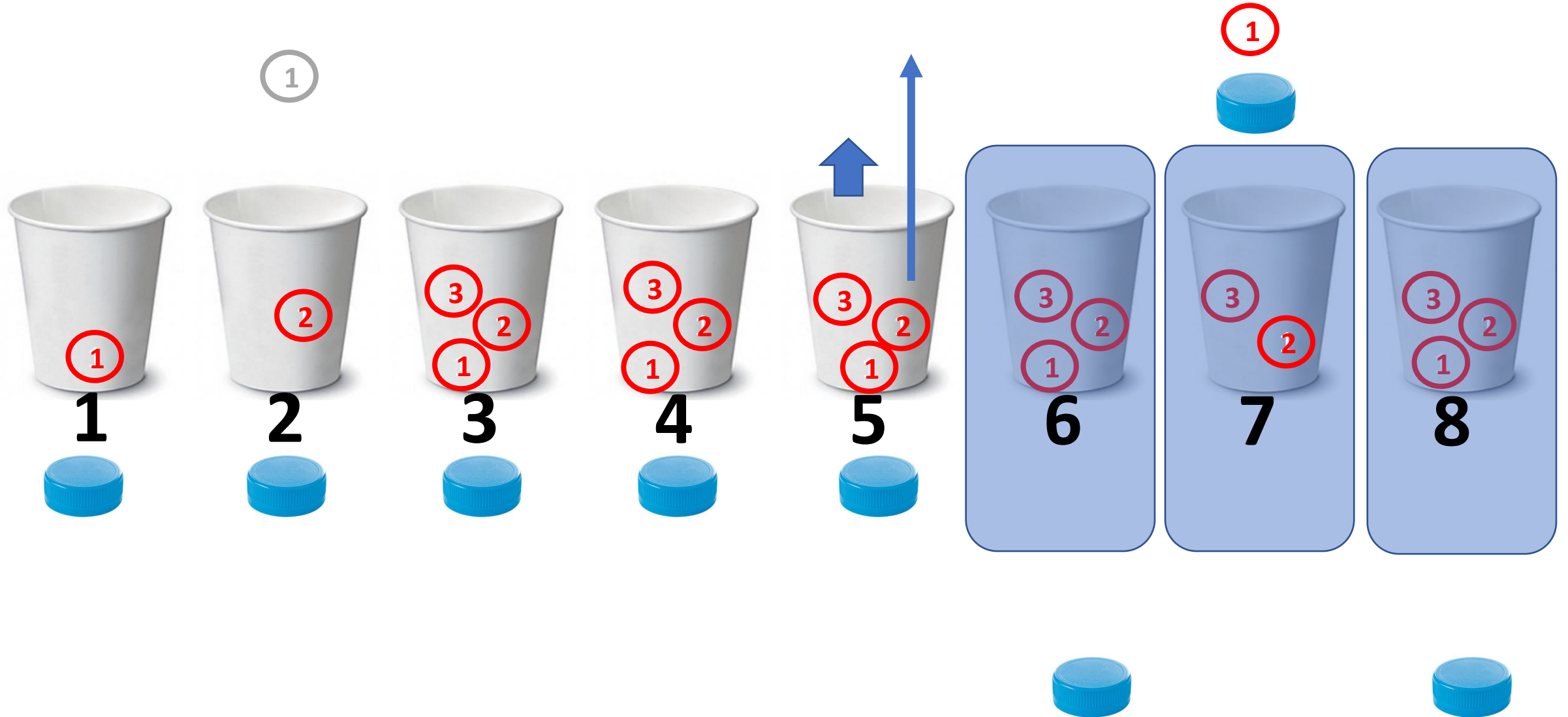
TOUR 1 : la machine joue. Elle tire au hasard le jeton **1**. Elle prend 1 bouchon.



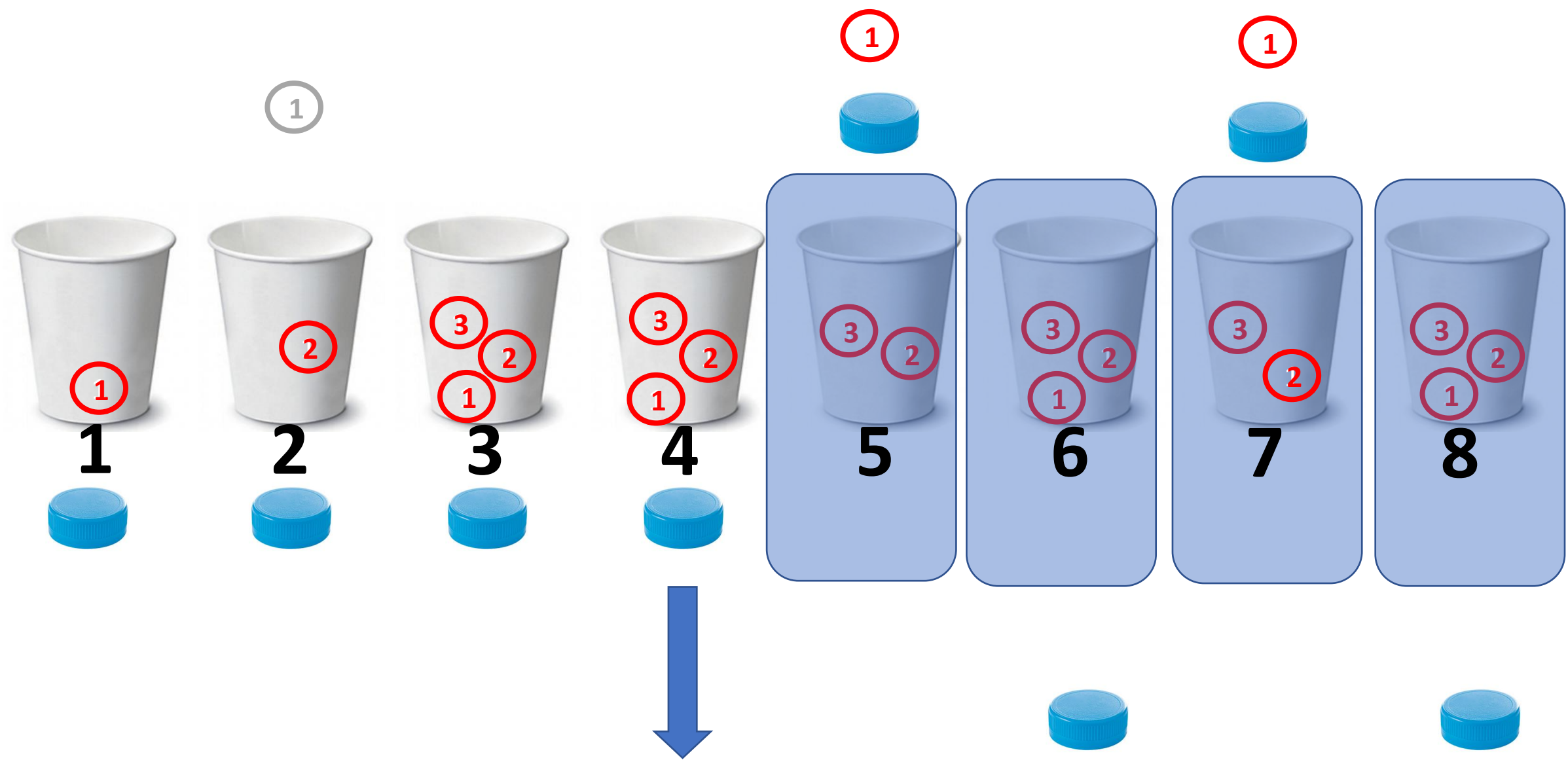
TOUR 2 : l'humain joue. Il prend 1 bouchon.



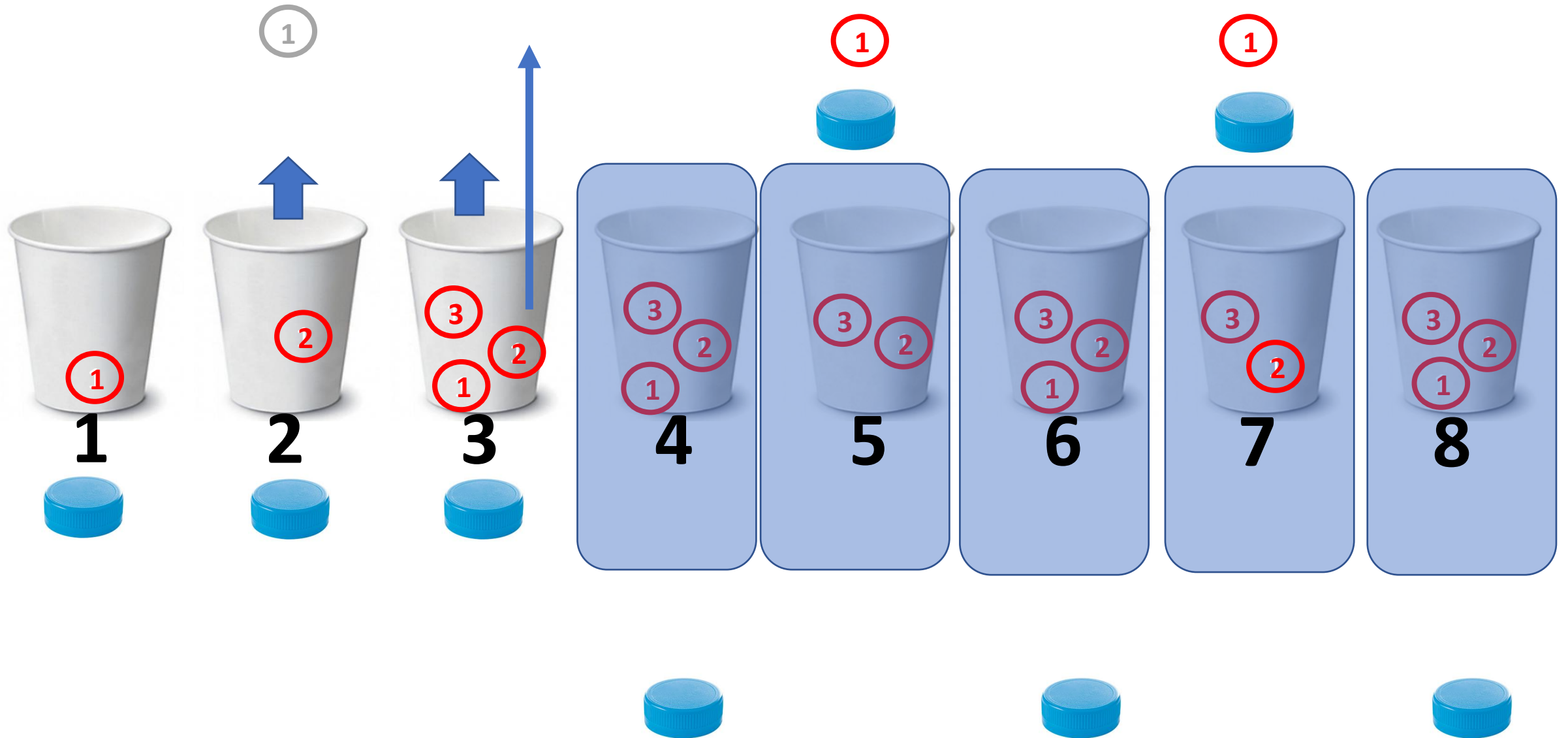
TOUR 2 : la machine joue. Elle tire au hasard le jeton ①. Elle prend 1 bouchon.



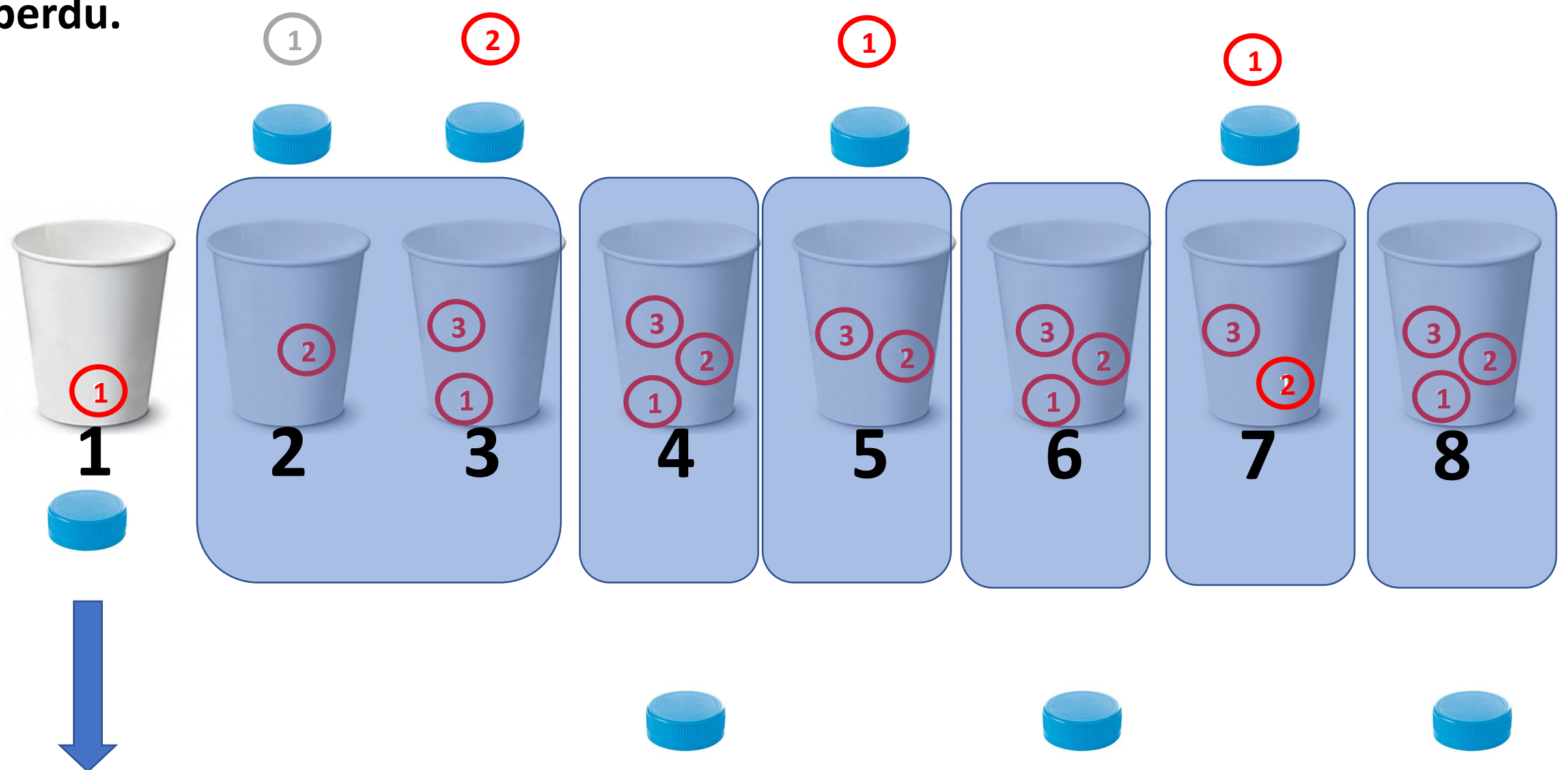
TOUR 3 : l'humain joue. Il prend 1 bouchon.



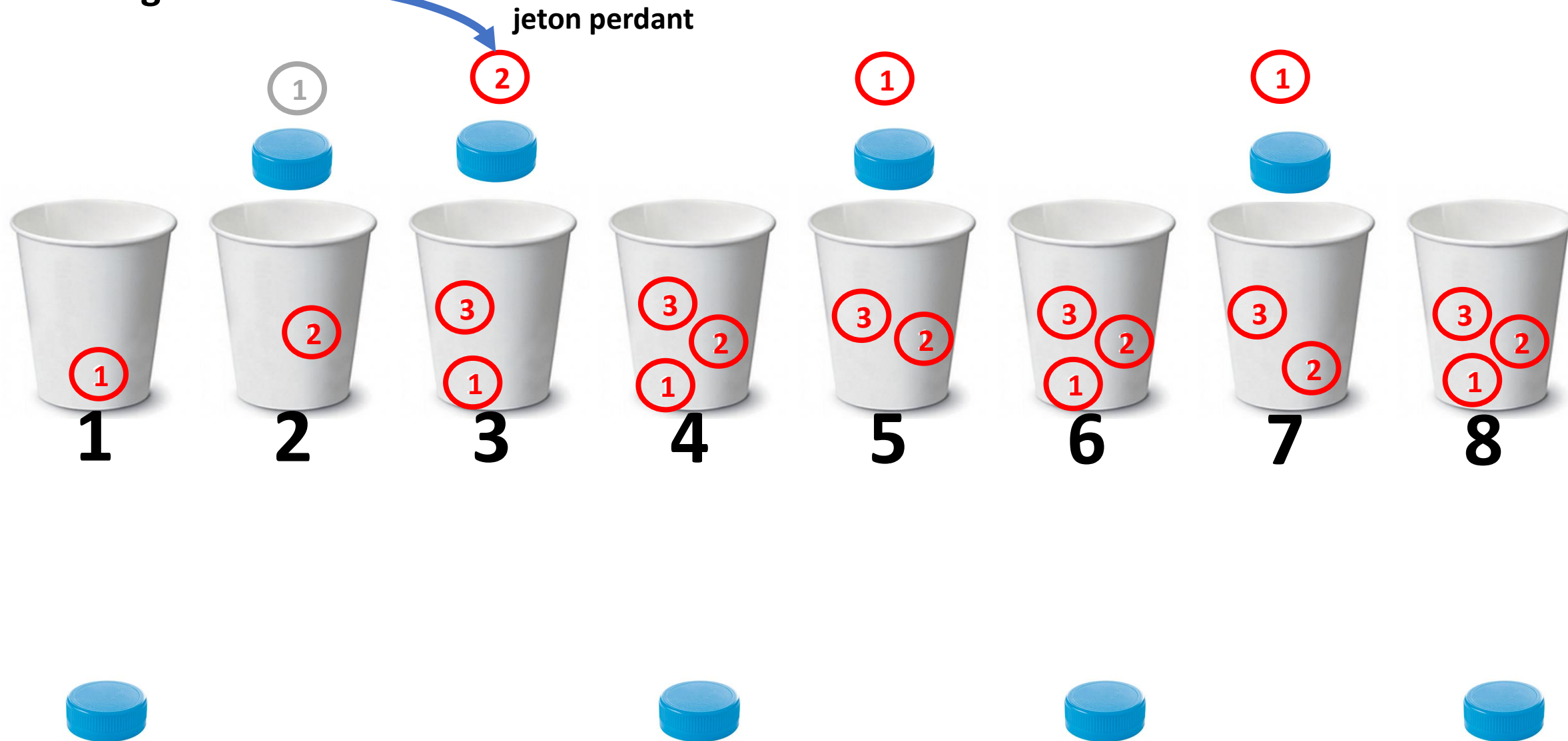
TOUR 3 : la machine joue. Elle tire au hasard le jeton **2**. Elle prend 2 bouchons.



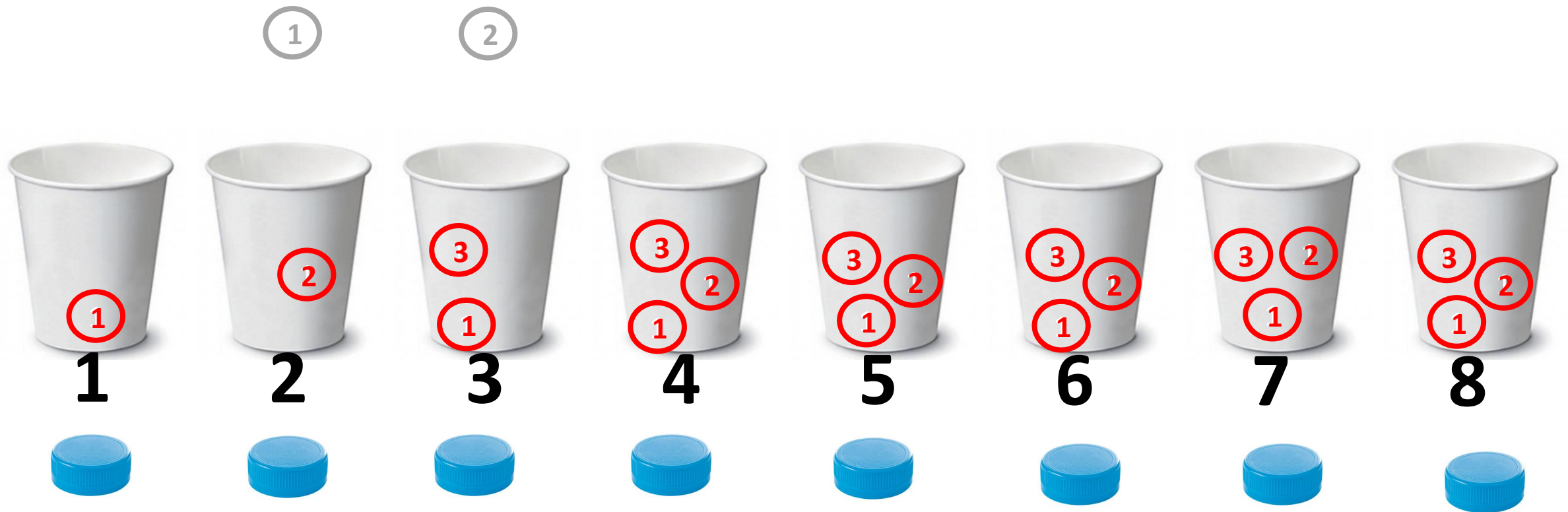
TOUR 4 : l'humain joue. Il prend le dernier bouchon. Il a gagné, la machine a perdu.



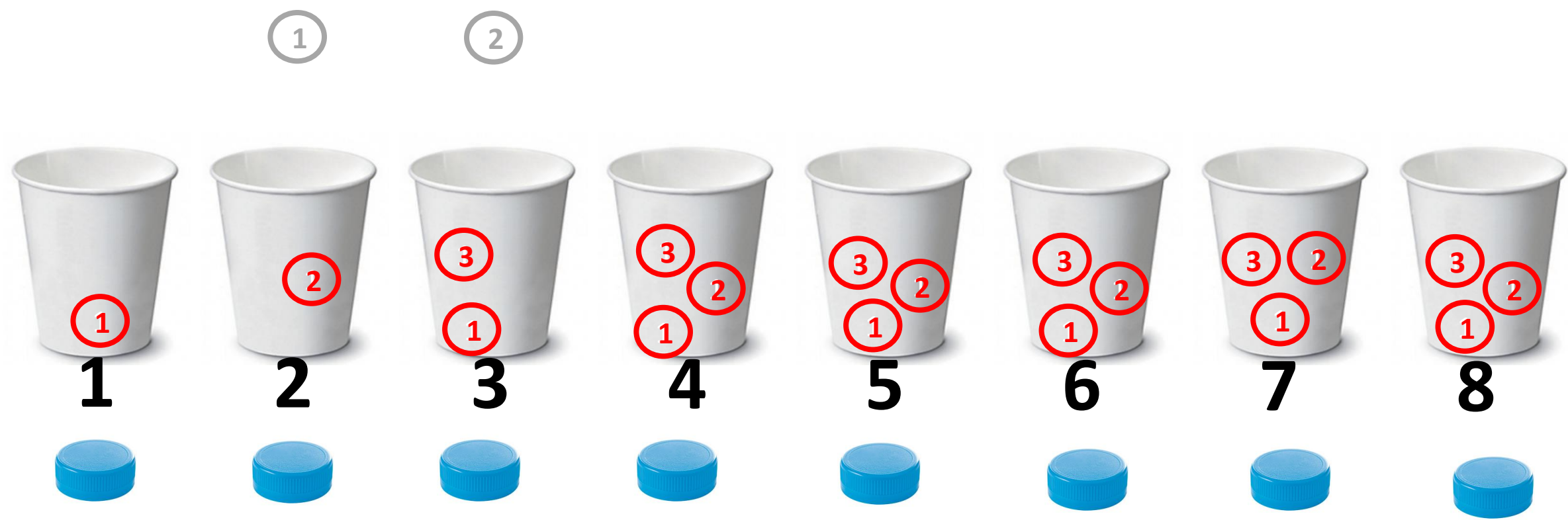
Fin de la 2^{ème} partie : la machine a perdu. Le dernier jeton tiré n'est pas remis dans le gobelet.



PARTIE 3 : SITUATION DE DEPART

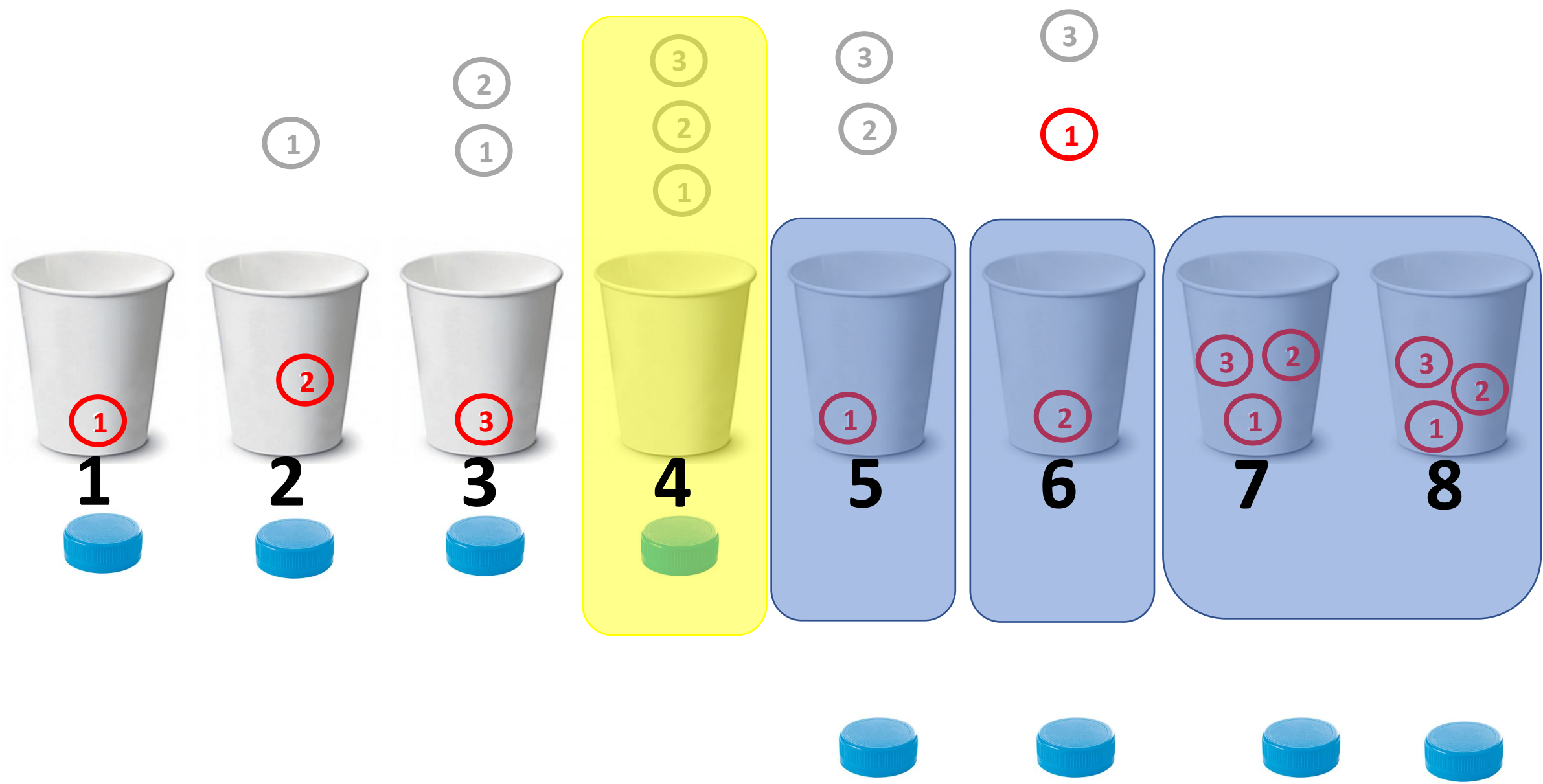


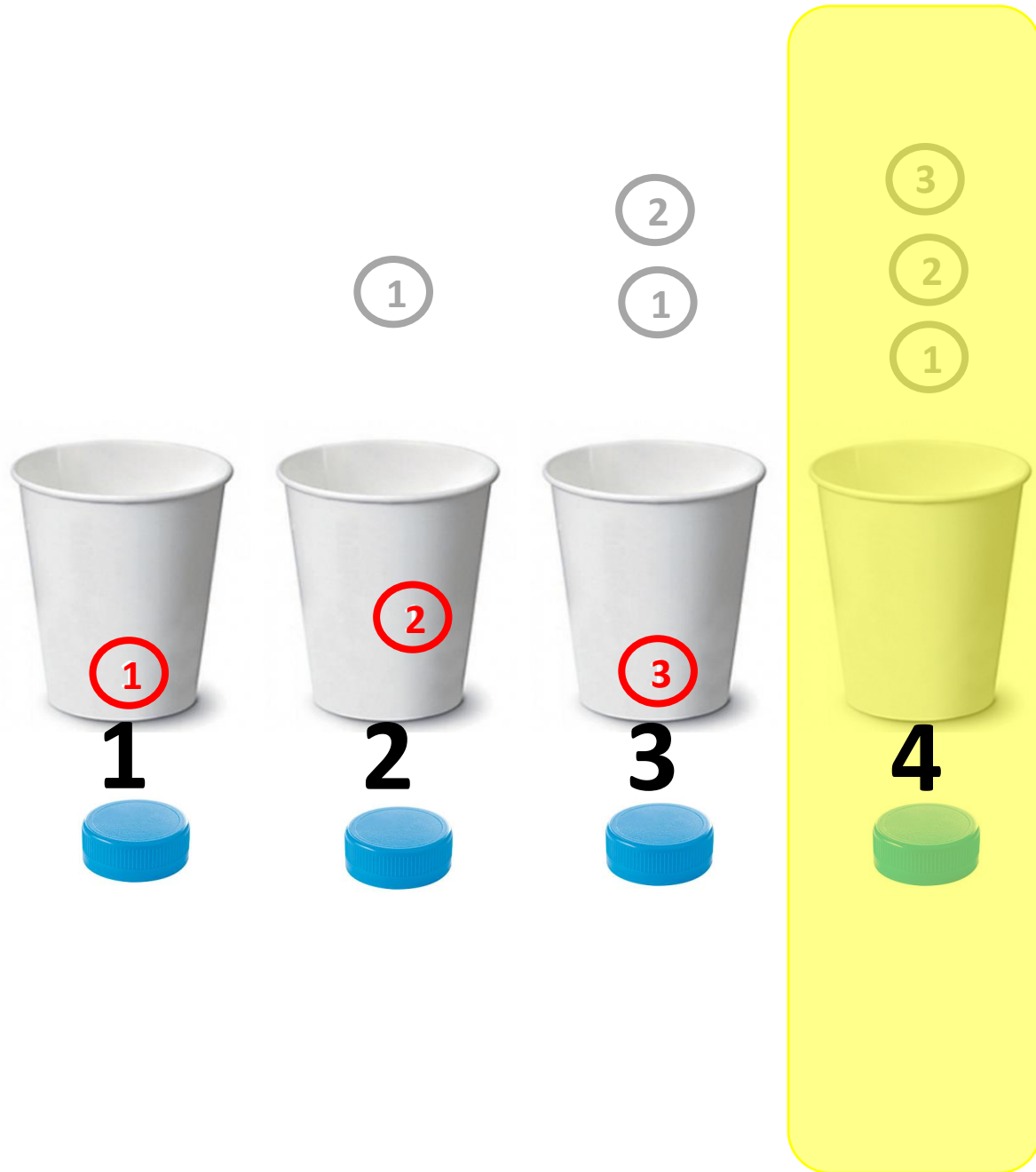
La machine va continuer à jouer. A chaque défaite, elle laisse de côté le dernier jeton joué. Elle élimine ainsi, quand elle perd, partie après partie, toutes les situations perdantes.





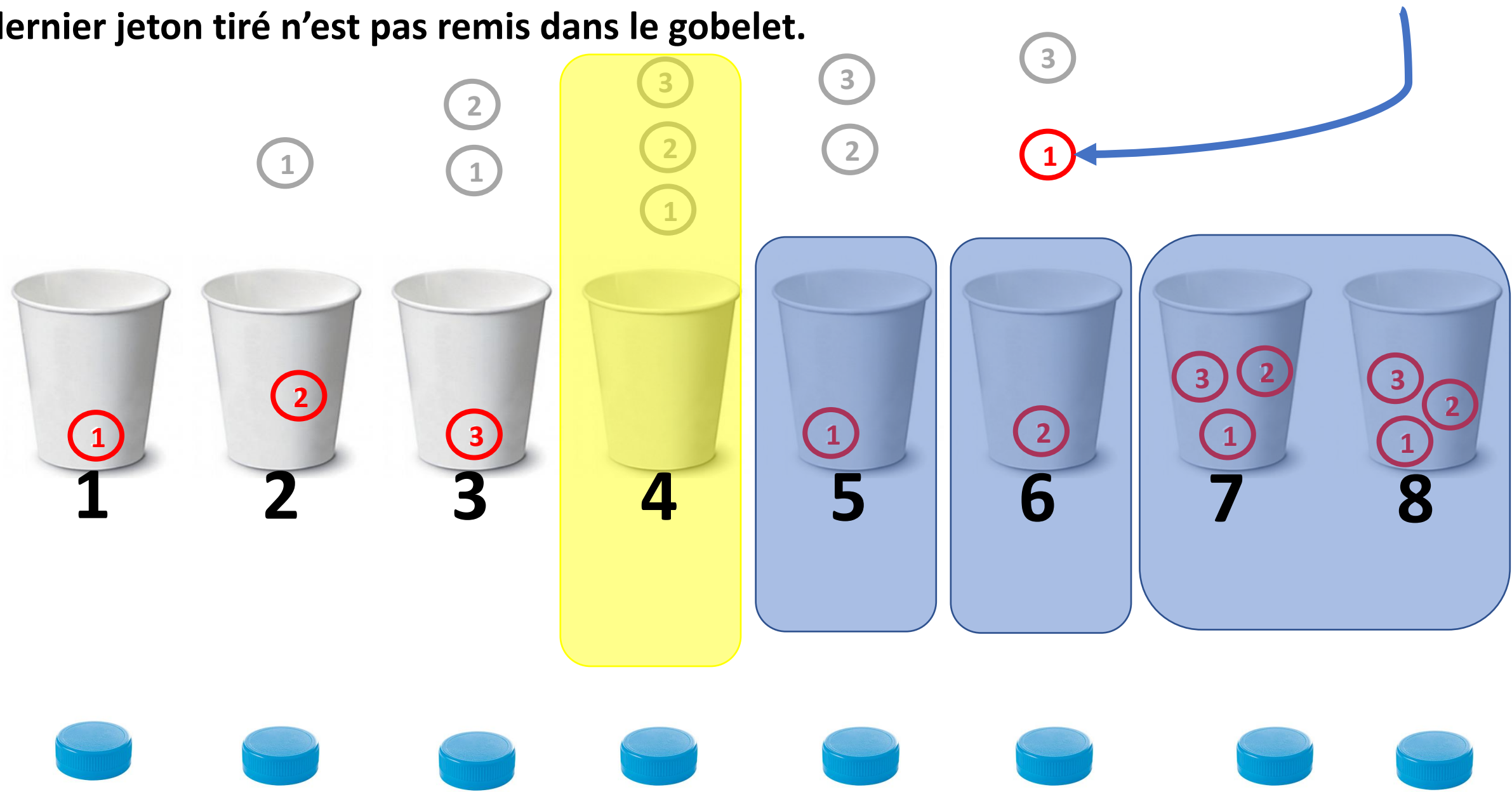
Que se passe-t-il dans ce cas précis (gobelet vide) ?



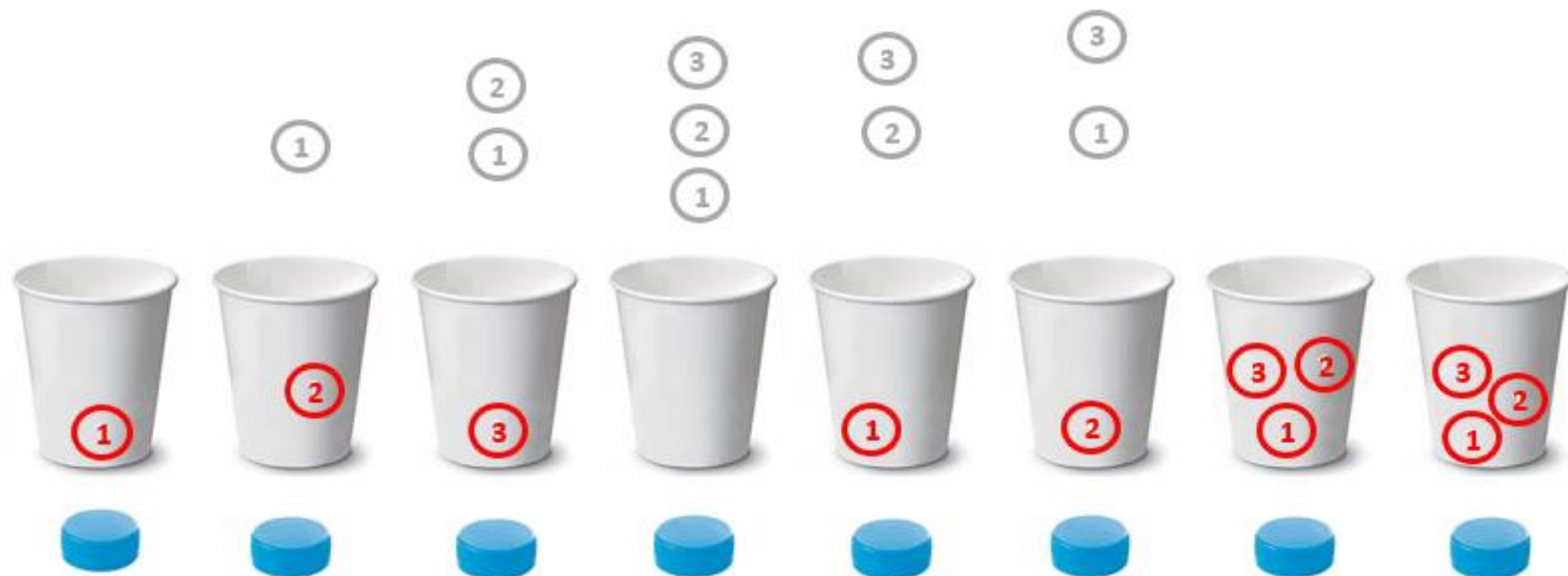


Le gobelet vide est considéré par la machine comme systématiquement perdant. En effet, quel que soit son choix, c'est perdu. Dès qu'elle tombe sur ce gobelet, la partie s'arrête, la machine a perdu.

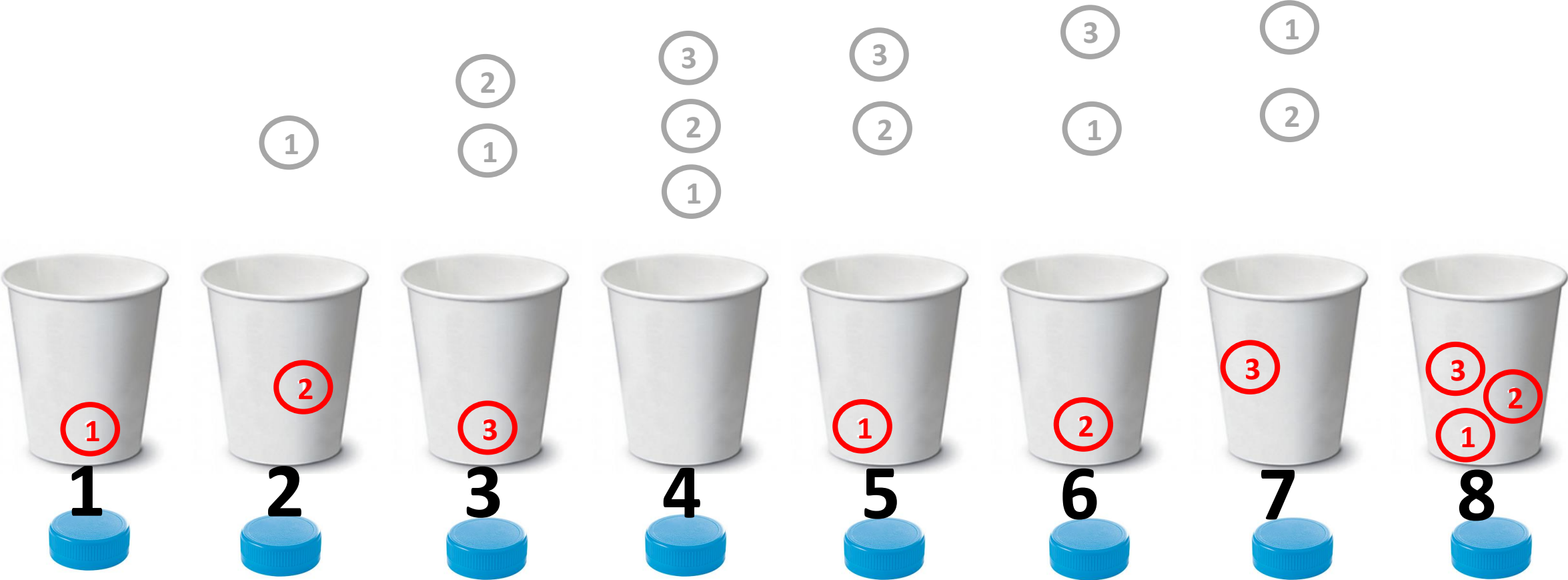
Fin de la 10^{ème} partie : la machine a perdu. Elle est tombée sur le gobelet vide. Le dernier jeton tiré n'est pas remis dans le gobelet.



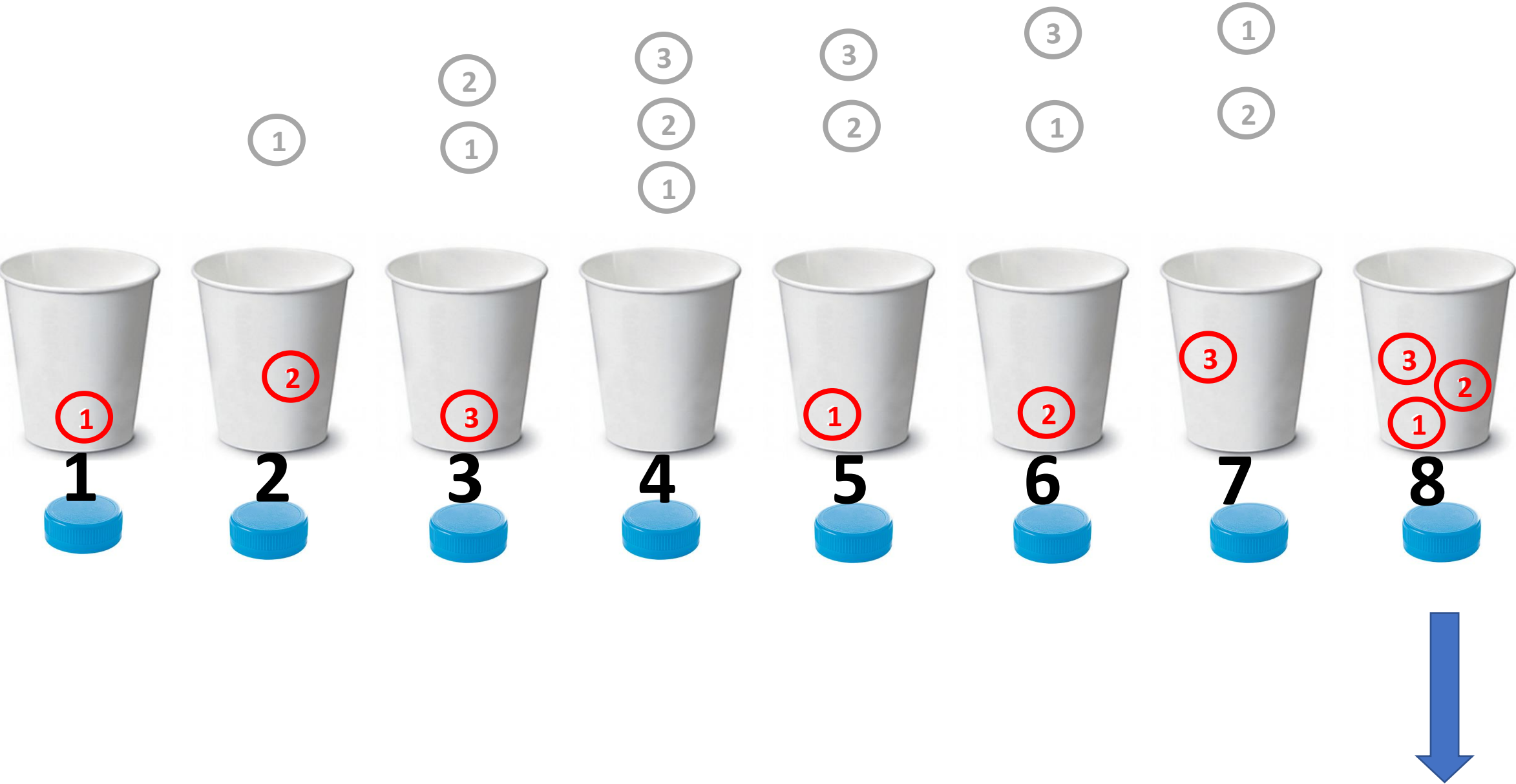
PARTIE 11 : SITUATION DE DEPART

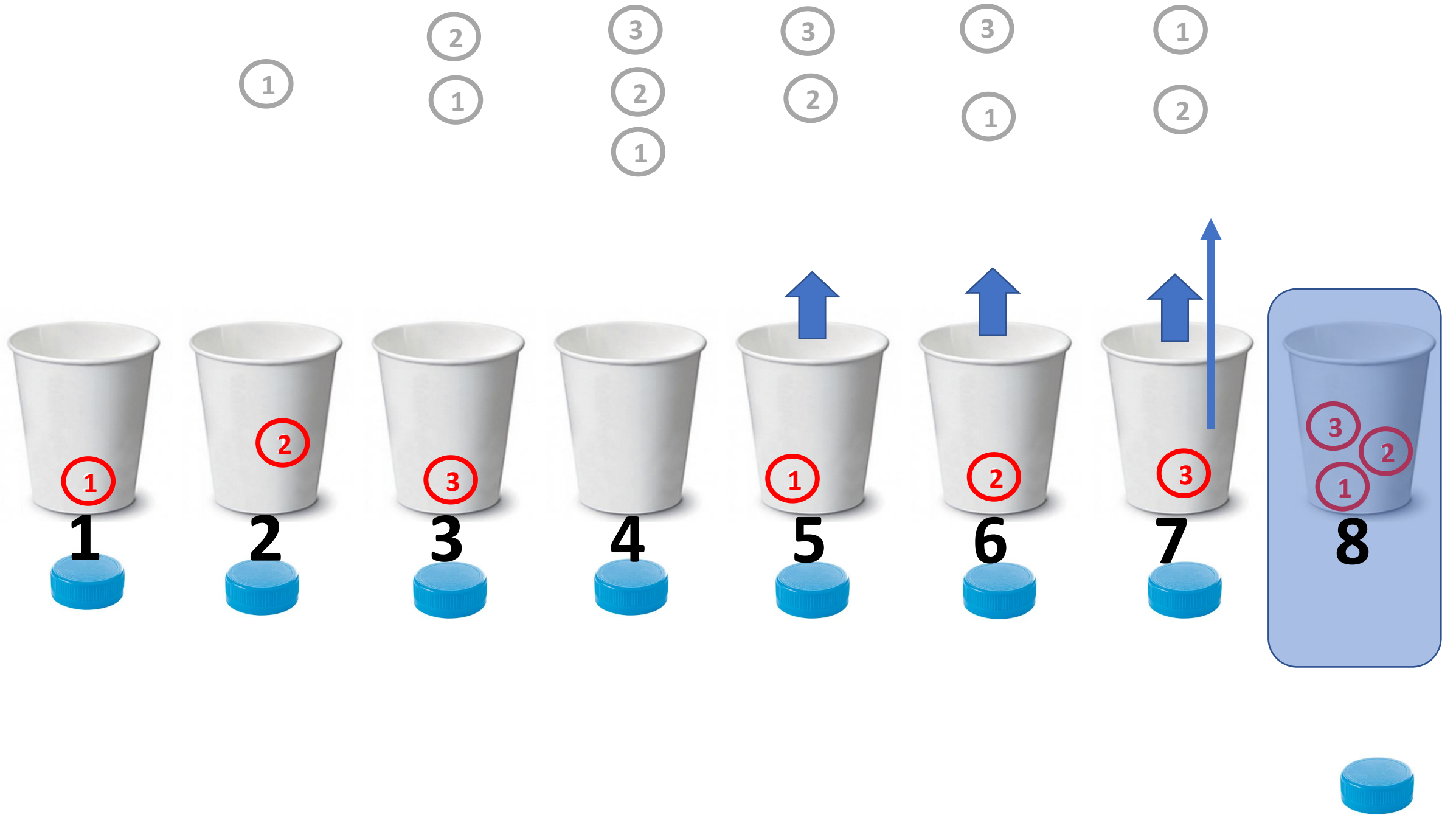


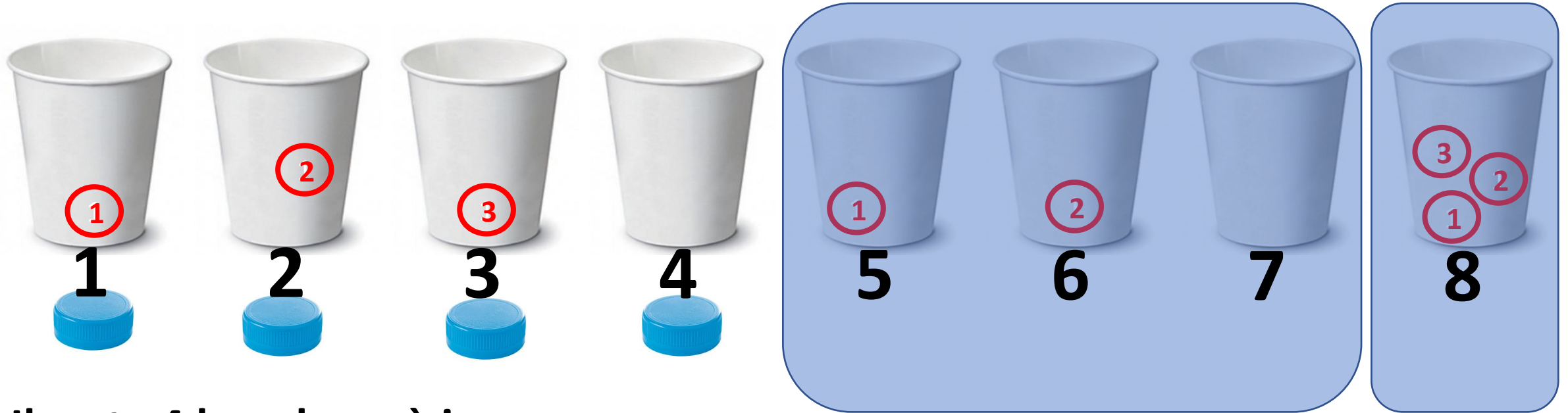
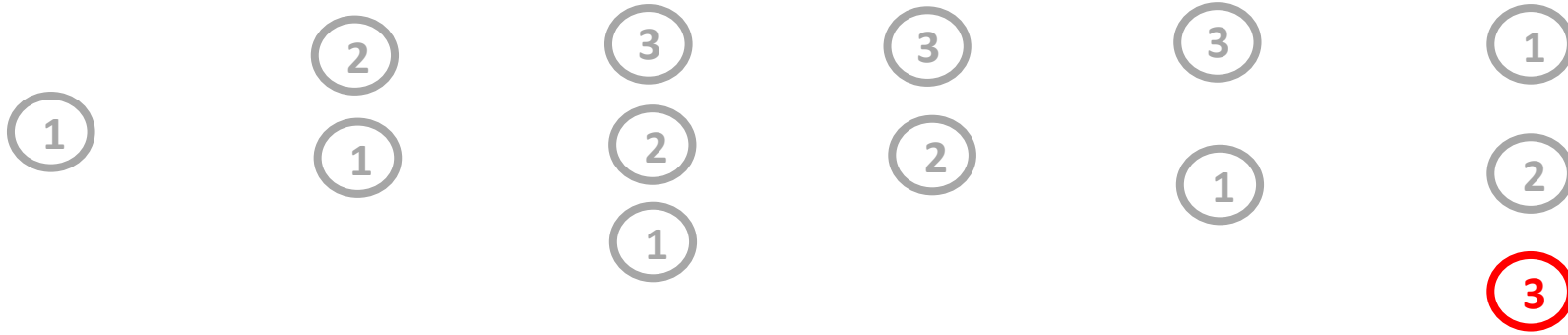
PARTIE 13 : SITUATION DE DEPART



CAS 1 : l'humain prend 1 bouchon.



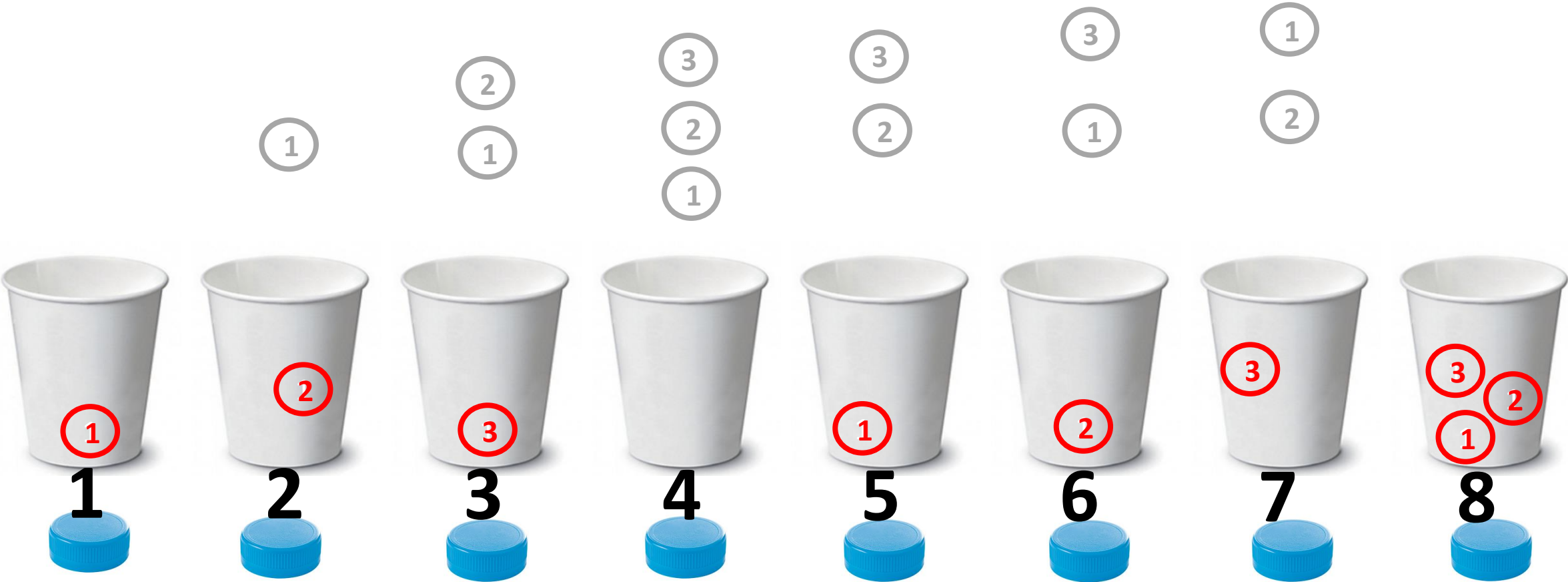




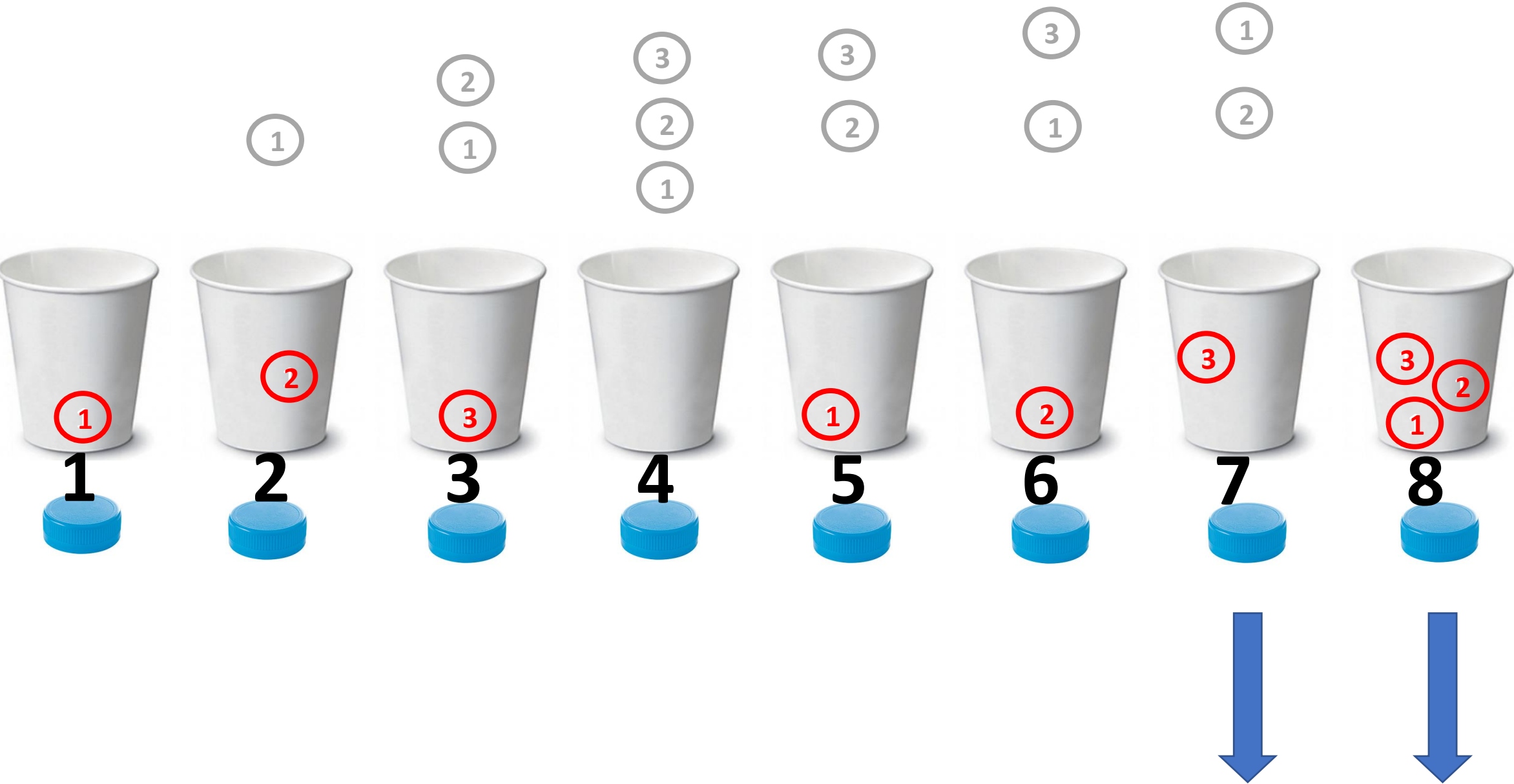
Il reste 4 bouchons à jouer :
l'humain a perdu dans tous les cas !
La machine gagne.

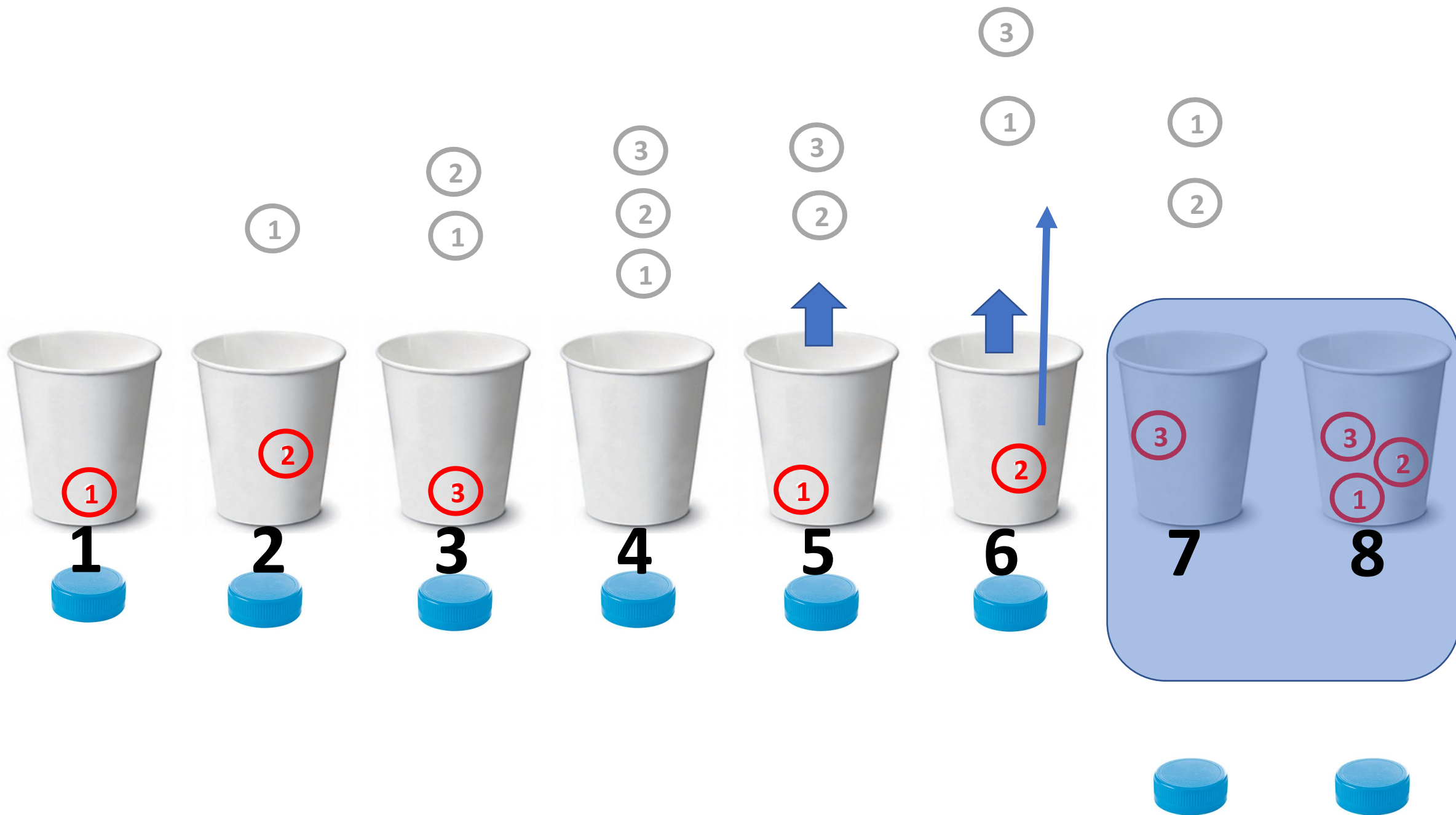


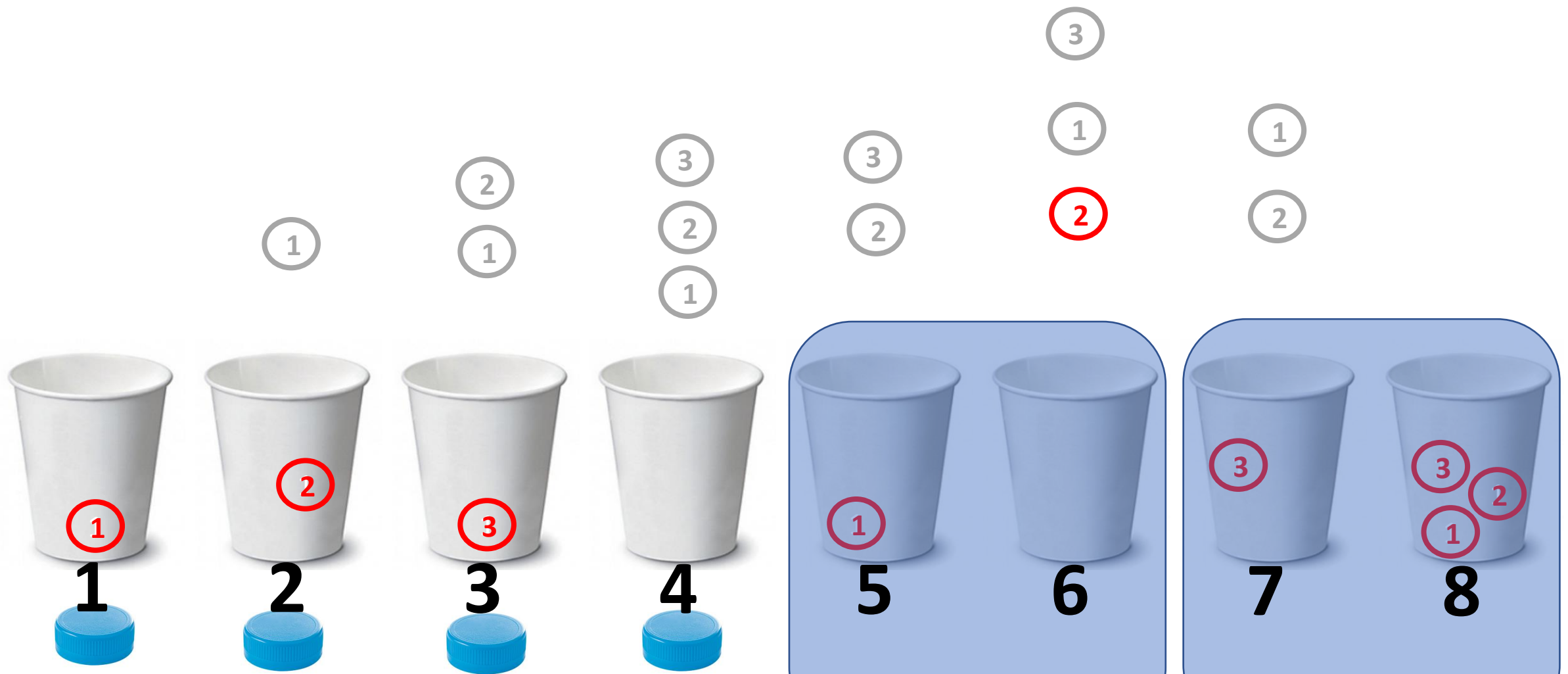
PARTIE 14 : SITUATION DE DEPART



CAS 2 : l'humain prend 2 bouchons.



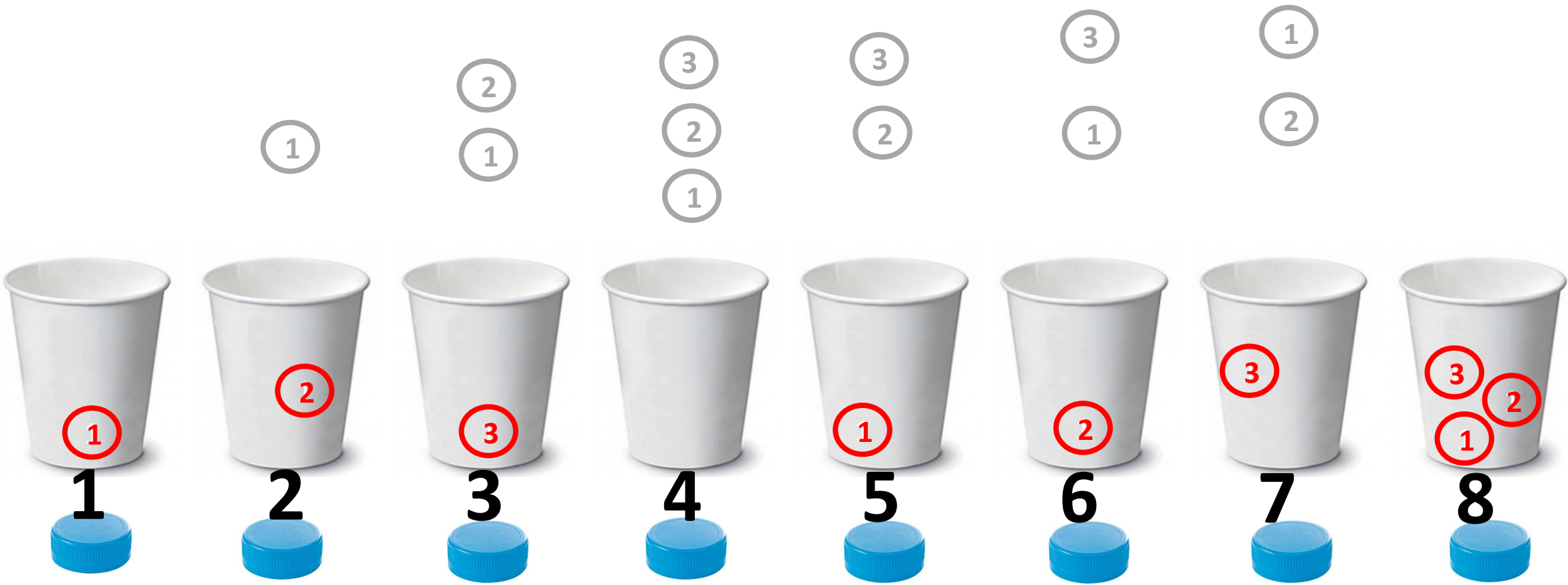




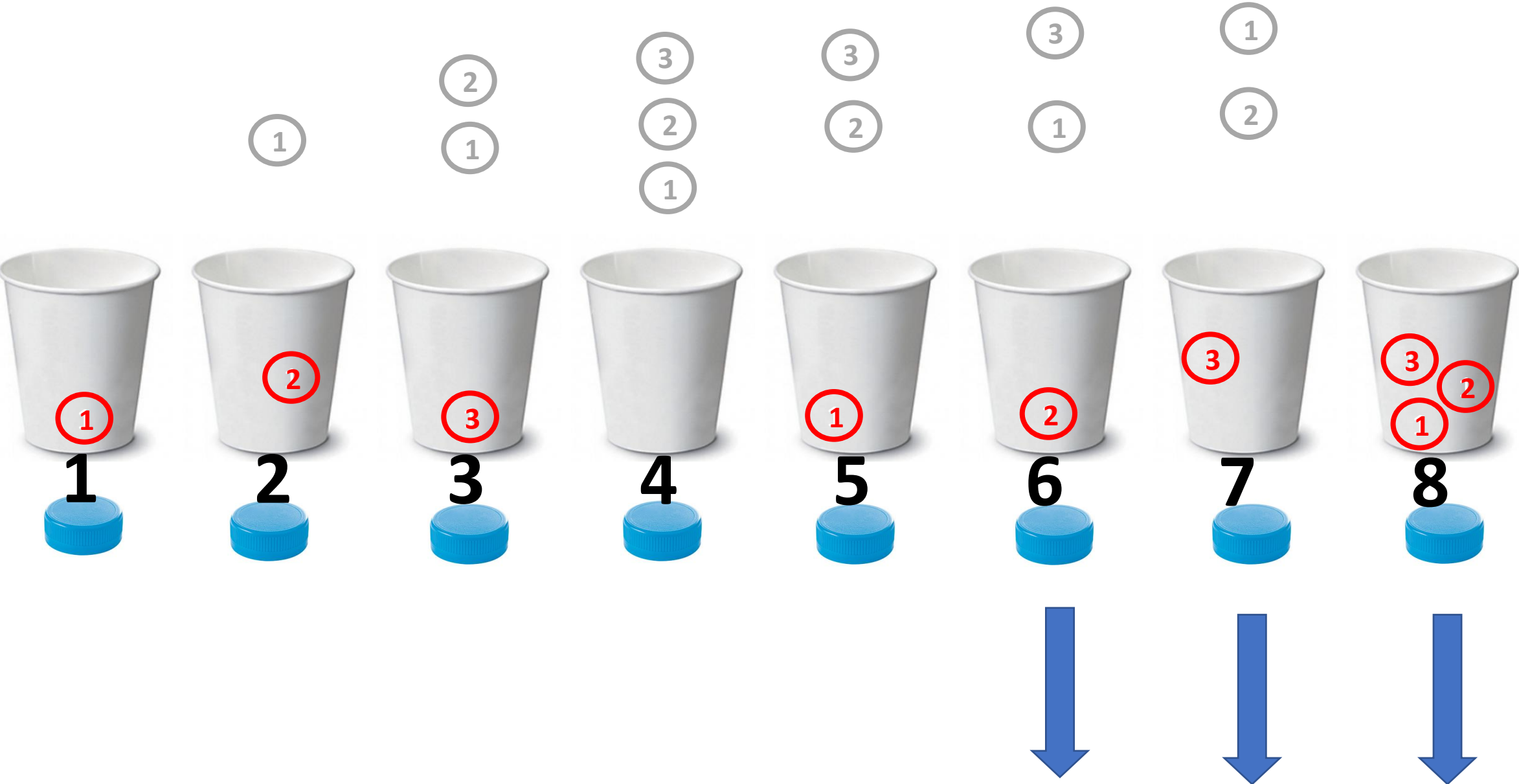
Il reste 4 bouchons à jouer :
l'humain a perdu dans tous les cas !
La machine gagne.

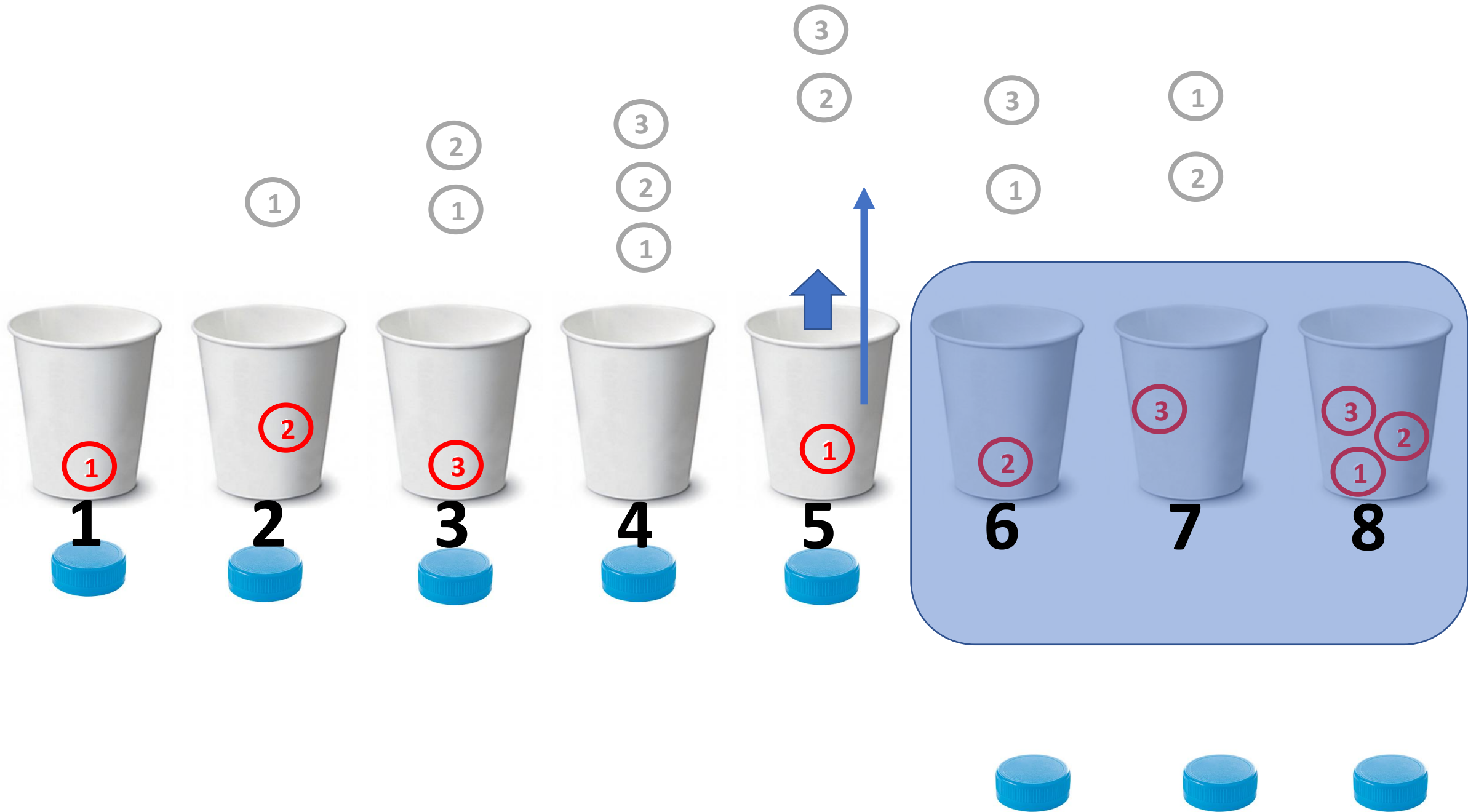


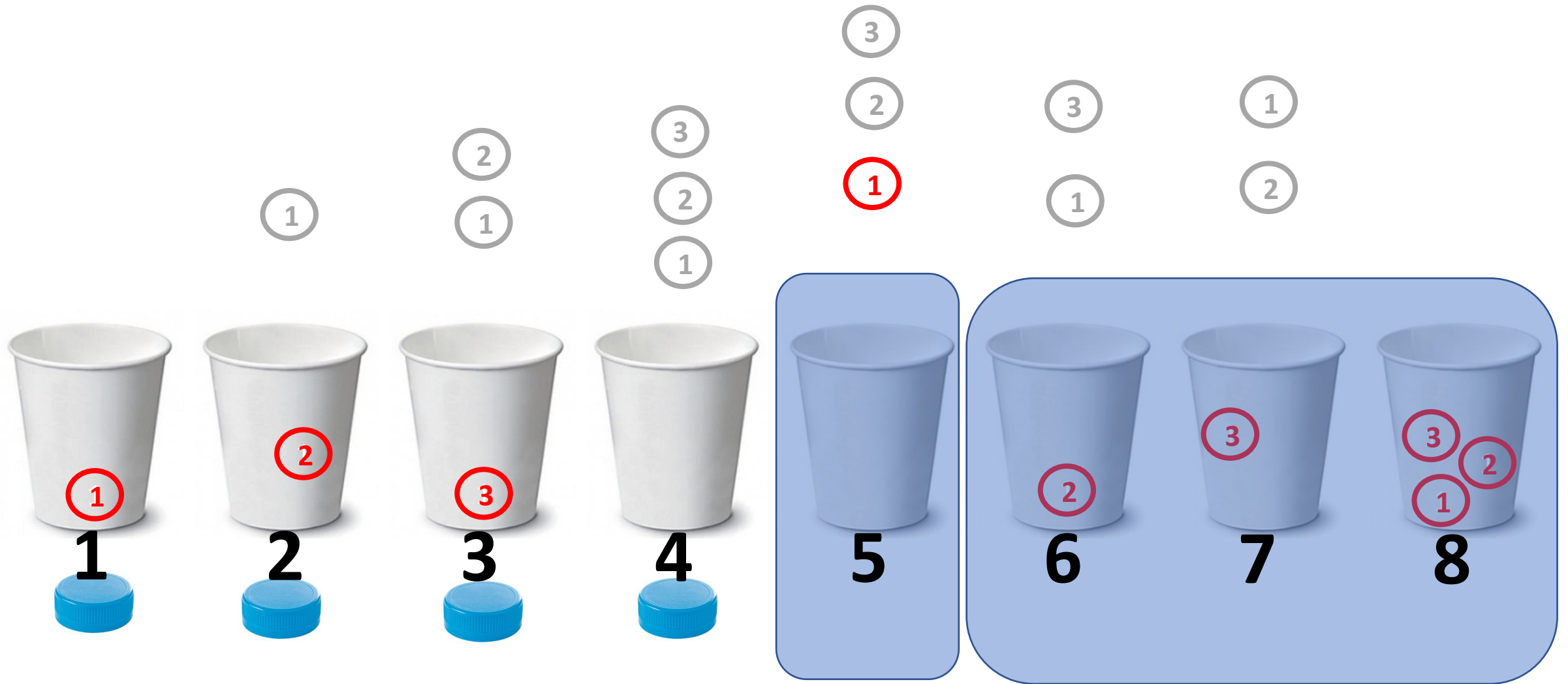
PARTIE 15 : SITUATION DE DEPART



CAS 3 : l'humain prend 3 bouchons.







Il reste 4 bouchons à jouer :
l'humain a perdu dans tous les cas !
La machine gagne.



Après un temps d'apprentissage, l'algorithme implanté dans la machine gagne systématiquement.

Il a appris par essais/erreur.

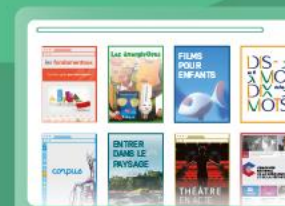
En IA, on parle d'apprentissage par renforcement.

Les ressources

- Enseigner et apprendre les sciences informatiques à l'école / Robert DI Cosmo) Interstices septembre 2015 https://interstices.info/jcms/c_47072/enseigner-et-apprendre-les-sciences-informatiques-a-l-ecole
- Computer Science Unplugged / Tim Bell, Ian H. Witten et Mike Fellows <http://csunplugged.org/activities/> et sa traduction française https://interstices.info/upload/csunplugged/CSUnplugged_fr.pdf
- La page de Marie Duflot <https://members.loria.fr/MDuflot/>
- La page de Martin Quinson (avec entre autres des docs, le matériel et des vidéos sur le crêpier, Nim, la planche à clous,...)

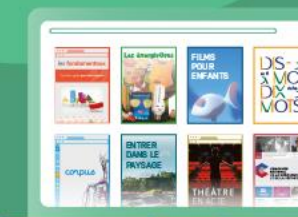
<http://people.irisa.fr/Martin.Quinson/Mediation/SMN/>

- La playlist Youtube Inria/Pixees des activités d' info sans ordi : <https://www.youtube.com/playlist?list=PLWvGMqXvyJAPSMFgCiy6qVHW9bAPu93X5>

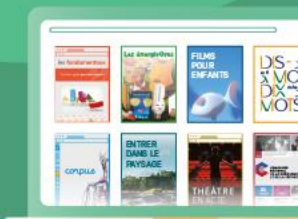


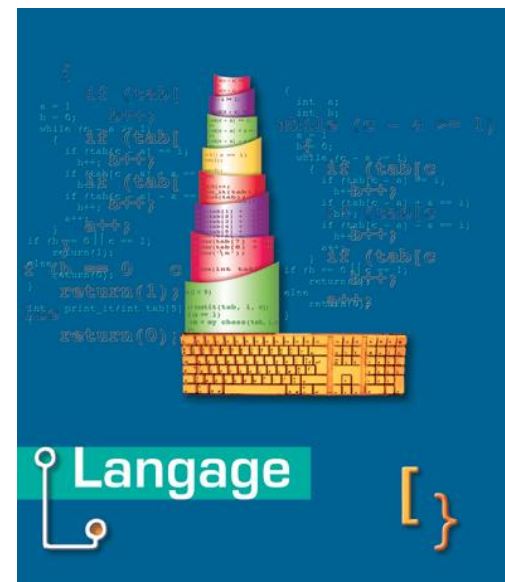
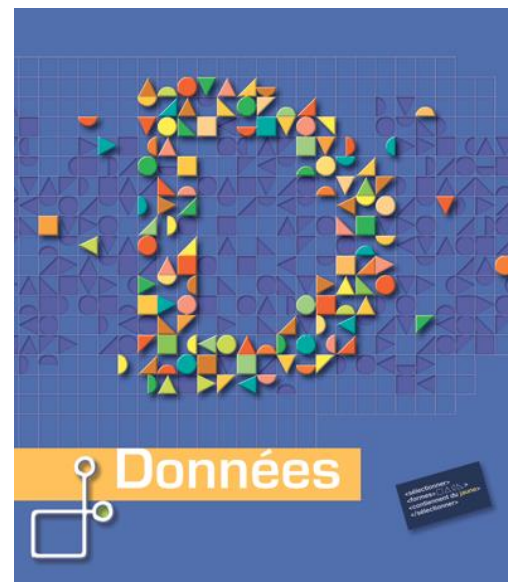
Les ressources

- Le parcours Class'Code activités débranchées <https://pixees.fr/classcode-v2/a-la-carte/>
- Le réseau national "informatique sans ordinateur" <https://www.societe-informatique-de-france.fr/mediation/infosansordi/>
- Du côté des IREM : IREM de Clermont Ferrand [<http://www.irem.univ-bpclermont.fr/Informatique-sans-ordinateur-IREM>], de Grenoble <http://www-irem.ujf-grenoble.fr/spip/spip.php?article146>
- Mais aussi côté Université de Poitiers <https://framagit.org/poitiers-infosansordi>
- L'informatique débranchée, n° 42-43 de Tangente éducation, 2018 <http://www.tangente-education.com/sommaire.php?som=271>
- <https://pixees.fr/tag/activite-debranchee/> (Philippe Atelier 33)
- Sylvie Alayrangues, Samuel Peltier, Laurent Signac. [Informatique débranchée: construire sa pensée informatique sans ordinateur.](#) Colloque Mathématiques en Cycle 3 IREM de Poitiers, IREM de Poitiers, Jun 2017, Poitiers, France. pp.216-226. hal-01868132
- [Que disent les sciences de l'éducation à propos de l'apprentissage du code ?](#) Margarida Romero, Stéphanie Noirpoudre, Thierry Viéville



Retrouvez nos ressources sur
reseau-canope.fr/canotech





CRÉDITS

- **Frédéric Dupuy**, frederic.dupuy@reseau-canope.fr
Médiateur de ressource et services - formation - Atelier Canopé 33
- **Emmanuel Page**, emmanuel.page@reseau-canope.fr
Chef de projet « Territoires numériques éducatifs »
(TNE) Réseau Canopé