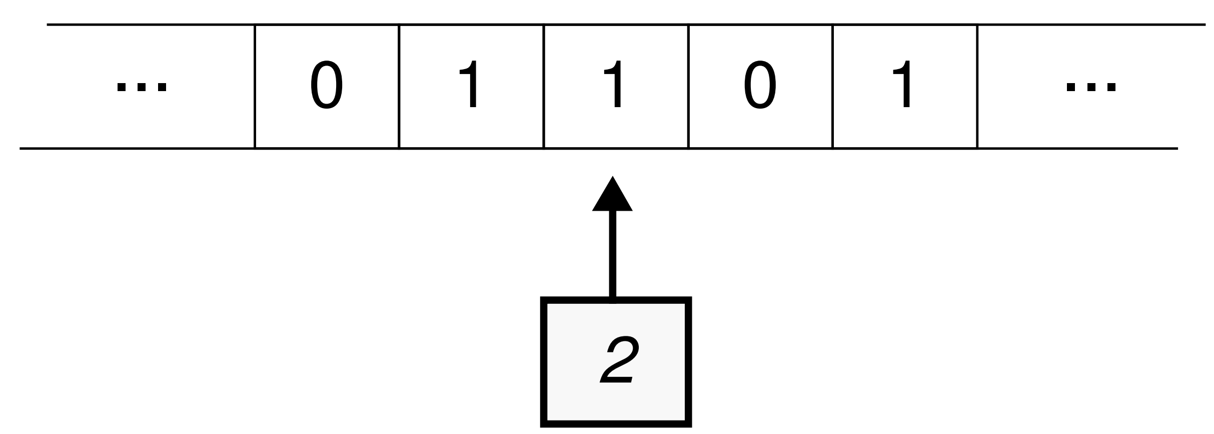
Machines de Turing : exemple

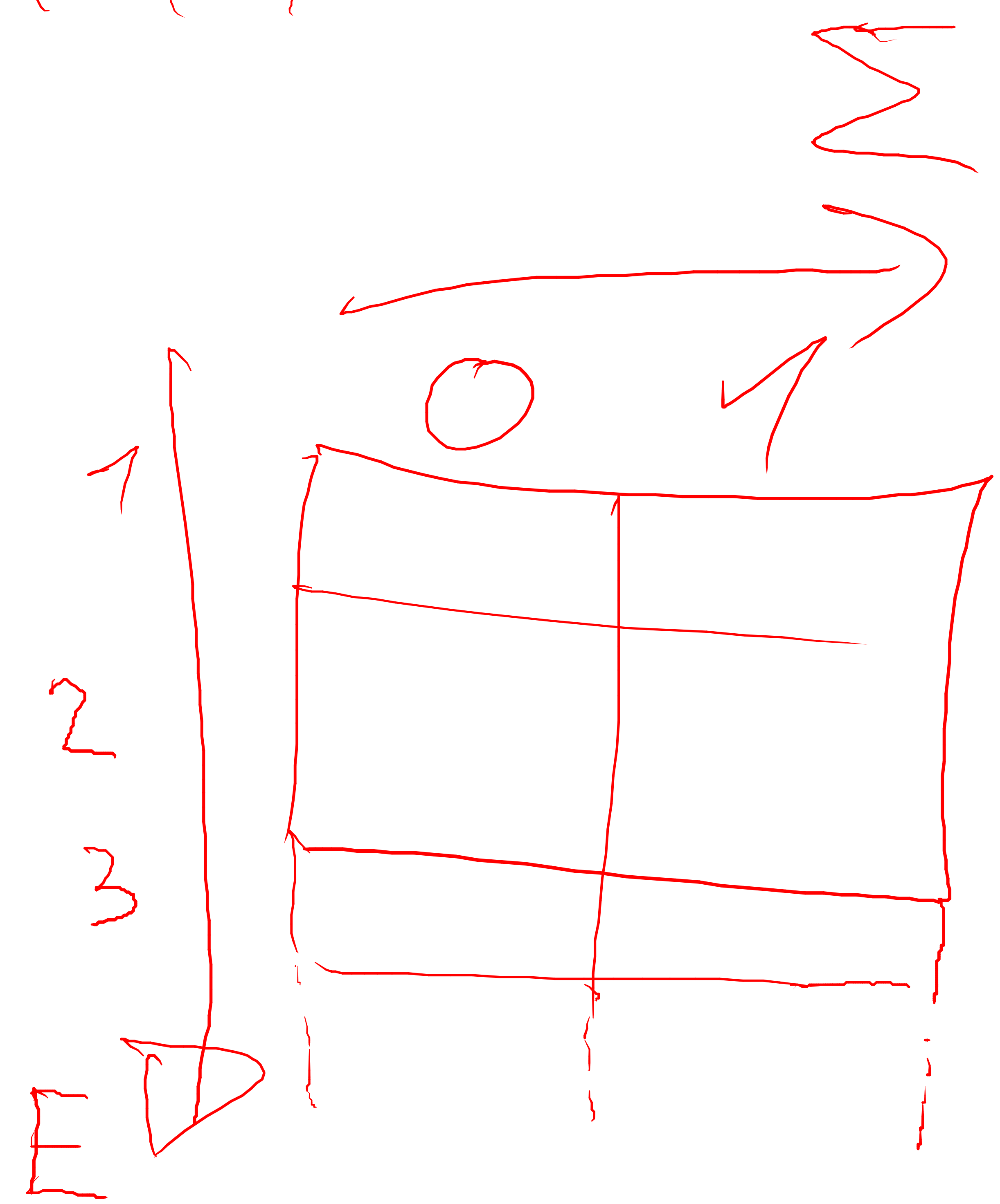
$$\Sigma = \{0, 1\}$$

$E = \{1, 2, 3\}$ (1 : état initial, 3 : état final)



état courant \ caractère courant	**0**	**1**	**ε**
1	(1,0,+)	(1,0,+)	(2,ε,-)
2	(2,0,-)	(2,0,-)	(3,ε,+)

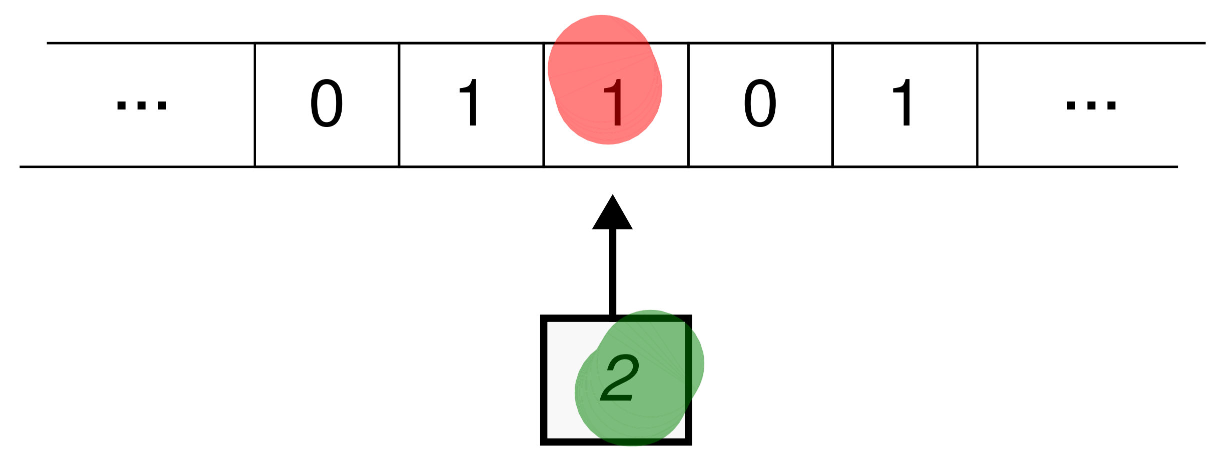
avec + (respect. -) indiquant un déplacement vers la droite (respect. vers la gauche).

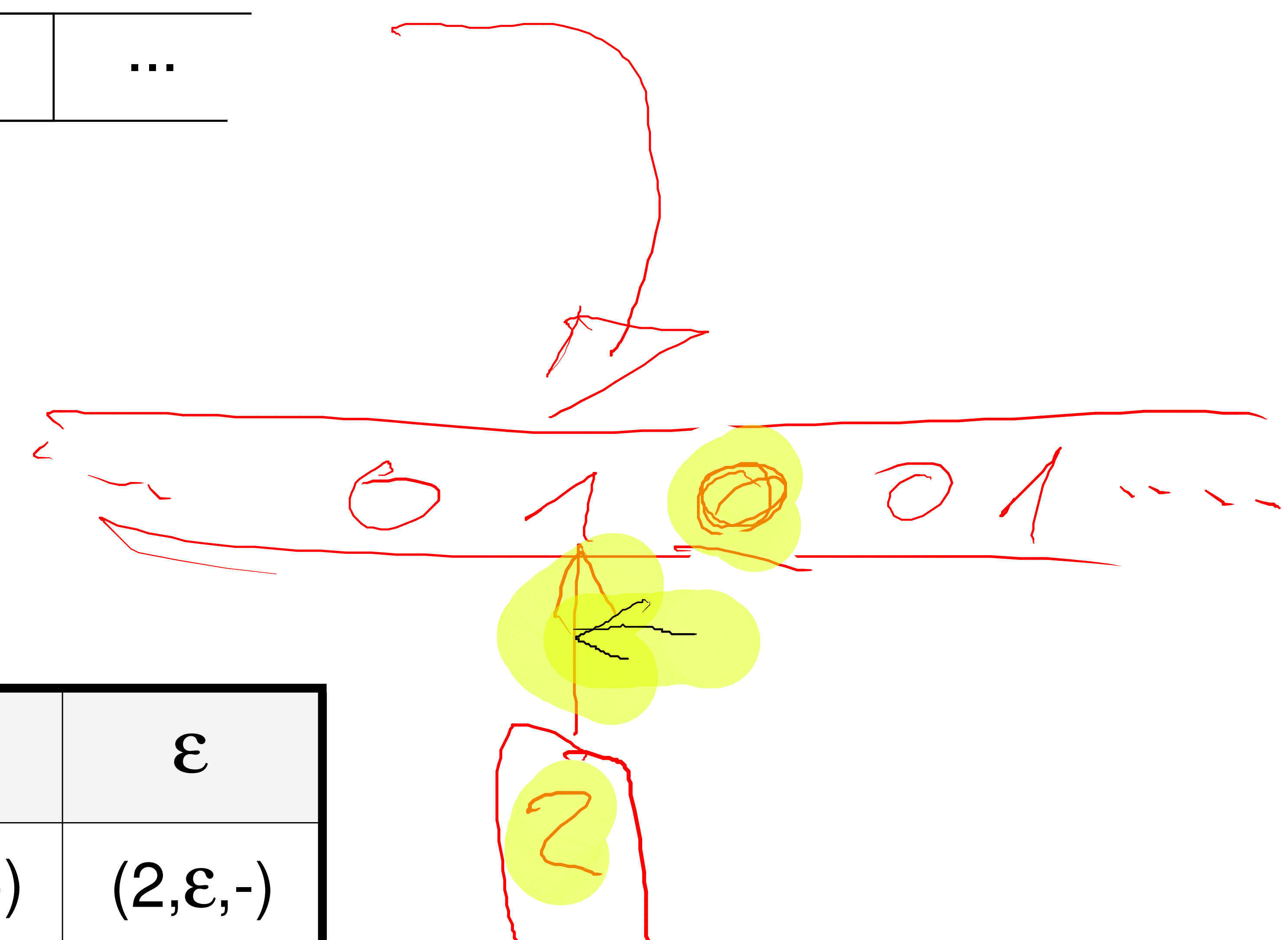


Machines de Turing : exemple

$$\Sigma = \{0, 1\}$$

$$E = \{1, 2, 3\} \qquad \text{(1 : état initial, 3 : état final)}$$





état courant \ caractère courant	**0**	**1**	**ε**
1	(1,0,+)	(1,0,+)	(2,ε,-)
2	(2,0,-)	(2,0,-)	(3,ε,+)

avec + (respect. -) indiquant un déplacement vers la droite (respect. vers la gauche).

Exemple : déterminer si un nombre est pair

Entrée : le nombre à tester, écrit en binaire
Sortie : 1 si le nombre est pair, 0 sinon

	0	1	ε
1	(1,0,+)	(1,1,+)	(2,ε,–)
2	(3,ε,-)	(4,ε,-)	(inutile)
3	(3,ε,-)	(3,ε,-)	(5,1,+)
4	(4,ε,-)	(4,ε,-)	(5,0,+)
5	(inutile)	(inutile)	(6,ε,-)

Exemple : entrée :

$$\cdots \varepsilon\ 1\ \cancel{0}\ \varepsilon \cdots$$

3

$\cdots \varepsilon\ 1\ \varepsilon\ \varepsilon \cdots$

$\boxed{6}$

Les paires d’entiers positifs sont dénombrables

« *Sûrement pas ! Après tout, il y a un nombre infini de paires qui ont toutes le même premier élément !* »

Et pourtant. . .

Ecrivons les paires sur un tableau comme ci-dessous :

1, 1	1, 2	1, 3	1, 4	1, 5	. . .
2, 1	2, 2	2, 3	2, 4	2, 5	. . .
3, 1	3, 2	3, 3	3, 4	3, 5	. . .
4, 1	4, 2	4, 3	4, 4	4, 5	. . .
5, 1	5, 2	5, 3	5, 4	5, 5	. . .

Deux paradoxes à 2’500 ans d’écart

Pouvons-nous donner des exemples concrets de fonctions booléennes qui ne peuvent pas être calculées ?

Oui, avec l’aide de deux paradoxes fameux :

- *Le paradoxe d’Epimenides (ou paradoxe du menteur)*
- *Le paradoxe de Berry*

Epimenides fut (peut-être) un philosophe Crétois qui aurait vécu il y a plus de 2’500 ans ; il aurait dit « tous les Crétois sont des menteurs ». Ceci n’est pas un vrai paradoxe (si Epimenides mentait, il n’y a pas de contradiction) ; il aurait plutôt du dire « je suis en train de vous mentir »

Ce paradoxe nous mène au problème de l’arrêt.

Berry fut bibliothécaire à Oxford à la fin du 19e. Lui aussi avait mal exprimé son paradoxe ; cela fut corrigé par le célèbre mathématicien Bertrand Russell :

« soit n le plus petit entier positif qui ne peut pas être défini en moins de ~~20~~ mots » n

un texte de ~~16~~ mots. . . log(n)

Ce paradoxe nous mène au problème de la longueur minimale de description.

Pour i de 1 à 100

écrire 3;

333333333333333...

La longueur minimale de description

Une question de base pour la compression des messages est simplement : quelle est la taille minimale d'un message transmettant l'information désirée ?

Théorème : Il n'existe pas d'algorithme pour déterminer la longueur minimale d'un programme qui produise un message donné.

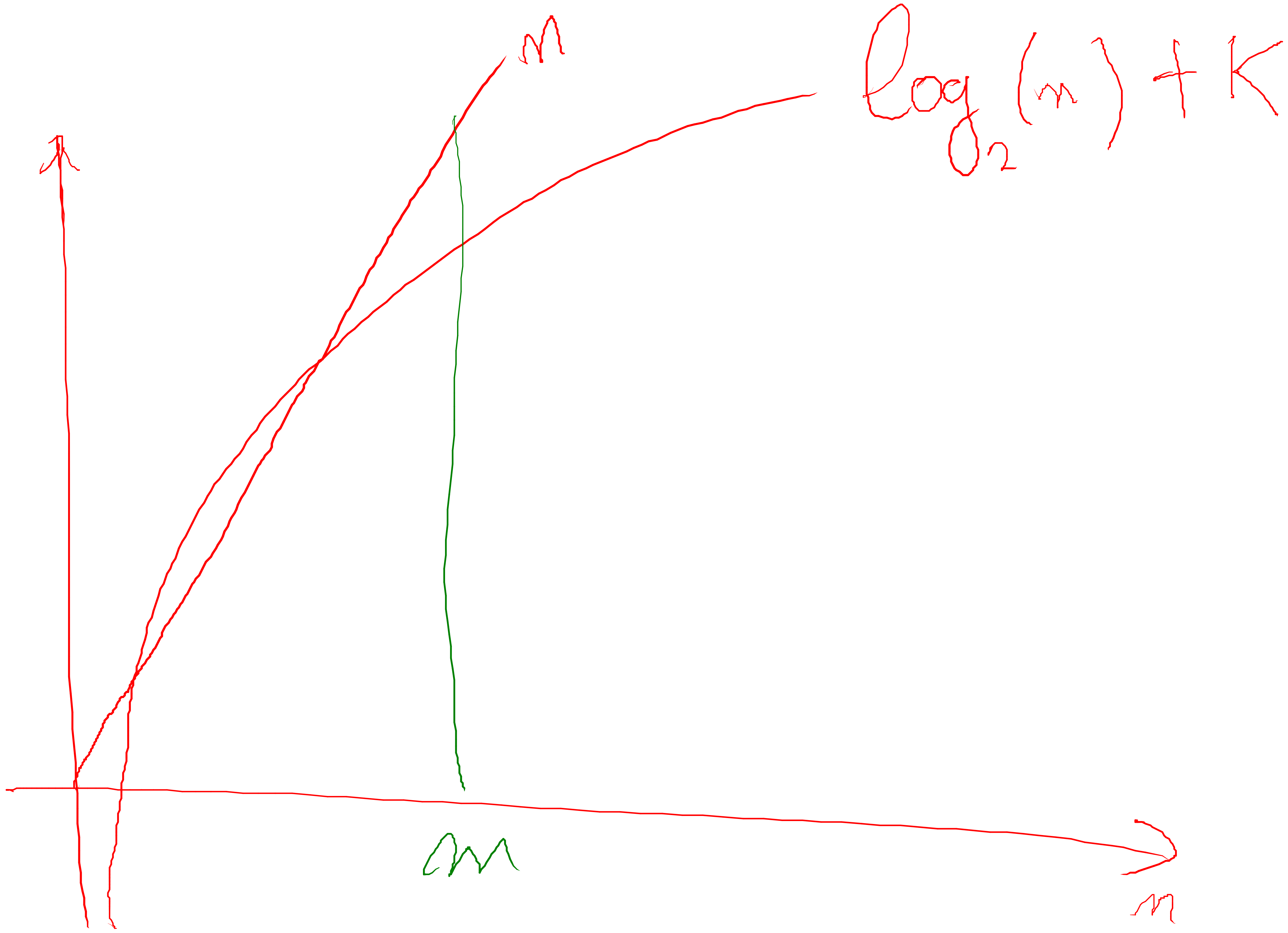
La preuve se fait par l'absurde. Disons donc qu'un tel algorithme existe et soit M un programme qui implémente cet algorithme. $M(n)$ retourne la longueur du programme le plus court qui puisse écrire n. Définissons le nouveau programme B comme suit (où m est un entier positif choisi) : K

B : $i \longleftarrow 0$; Répéter $i \longleftarrow i + 1$ tant que $M(i) \leq m$; Résultat : i

B retourne le plus petit entier i tel qu'aucun programme de longueur moindre que m ne puisse écrire i.

Quelle est la longueur de B ? Un nombre constant de caractères pour les instructions, plus les $\log_2 m$ caractères (chiffres) nécessaires pour encoder m en binaire. Pour m assez grand, la longueur de B est donc proche de $\log m$, ce qui est bien moindre que m.

B retourne le plus petit i tel qu'aucun programme de longueur moindre que m ne puisse écrire i, mais B lui-même écrit ce i et sa longueur est moindre que m : contradiction !



n
$\log_2(n) + K$
m
n

Résumé (à ce stade) de la classification des problèmes

difficulté

problème

Problèmes indécidables

Problèmes décidables

P

RÉSOUT

il existe algo

$O(n^k)$

n = taille entrée

Et Prem, où est-il ?

Rappel : Prem(n) :

entrée : n entier naturel (≥ 2)

question : n est-il premier ?

binaire 01102

$T = \log_2(n)$

T : taille de l'entrée

Une solution proposée :

1. si n est pair, il n'est premier que s'il vaut 2
2. sinon, tester la division de n par tous les entiers impairs plus petits que $\sqrt{n}$

$O(\sqrt{n})$

Cet algorithme est... ...*exponentiel ! !*

☞ **Attention !** à la **taille** de l'entrée ! ! !

$= O(2^{T/2})$