$1 + 1 = 0$ ou comment détecter les erreurs ?

Nicolas Billerey et François Martin

Laboratoire de Mathématiques
Université Blaise Pascal – Clermont-Ferrand 2

Fête de la science 2015

Sommaire

Quelques rappels...

Quelques rappels...

Nombres pairs $= \{\ldots, -6, -4, -2, 0, 2, 4, 6, \ldots\}$

Quelques rappels...

Nombres pairs $= \{\ldots, -6, -4, -2, 0, 2, 4, 6, \ldots\}$
Nombres impairs $= \{\ldots, -5, -3, -1, 1, 3, 5, \ldots\}$

Quelques rappels...

nombre pair

Quelques rappels...

nombre pair $+$ nombre pair $=$

Quelques rappels...

nombre pair $+$ nombre pair $=$ nombre pair

Quelques rappels...

$$\begin{array}{r} \text{nombre pair} \\ \text{nombre impair} \end{array} + \text{nombre pair} = \text{nombre pair}$$

Quelques rappels...

$$\begin{aligned} \text{nombre pair} &+ \text{nombre pair} &&= \text{nombre pair} \\ \text{nombre impair} &+ \text{nombre pair} &&= \end{aligned}$$

Quelques rappels...

$$\begin{aligned} \text{nombre pair} &+ \text{nombre pair} &&= \text{nombre pair} \\ \text{nombre impair} &+ \text{nombre pair} &&= \text{nombre impair} \end{aligned}$$

Quelques rappels...

$$\begin{array}{rrl} \text{nombre pair} & + \text{ nombre pair} & = \text{nombre pair} \\ \text{nombre impair} & + \text{ nombre pair} & = \text{nombre impair} \\ \text{nombre impair} & & \end{array}$$

Quelques rappels...

$$\begin{aligned} \text{nombre pair} &+ \text{nombre pair} &&= \text{nombre pair} \\ \text{nombre impair} &+ \text{nombre pair} &&= \text{nombre impair} \\ \text{nombre impair} &+ \text{nombre impair} &&= \end{aligned}$$

Quelques rappels...

$$\begin{aligned} \text{nombre pair} &+ \text{nombre pair} &&= \text{nombre pair} \\ \text{nombre impair} &+ \text{nombre pair} &&= \text{nombre impair} \\ \text{nombre impair} &+ \text{nombre impair} &&= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + \text{nombre pair} &= \text{nombre pair} \\ \text{nombre impair} + \text{nombre pair} &= \text{nombre impair} \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= \text{nombre pair} \\ \text{nombre impair} + \text{nombre pair} &= \text{nombre impair} \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= 0 \\ \text{nombre impair} + \text{nombre pair} &= \text{nombre impair} \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= 0 \\ 1 + \text{nombre pair} &= \text{nombre impair} \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= 0 \\ 1 + 0 &= \text{nombre impair} \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= 0 \\ 1 + 0 &= 1 \\ \text{nombre impair} + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{aligned} 0 + 0 &= 0 \\ 1 + 0 &= 1 \\ 1 + \text{nombre impair} &= \text{nombre pair} \end{aligned}$$

Quelques rappels...

$$\begin{array}{rrr} 0 + & 0 = & 0 \\ 1 + & 0= & 1 \\ 1 + & 1= & \text{nombre pair} \end{array}$$

Quelques rappels...

$$\begin{array}{rrr} 0+ & 0= & 0 \\ 1+ & 0= & 1 \\ 1+ & 1= & 0 \end{array}$$

Cherchez l'erreur !

Cherchez l'erreur !

$$1 + 1 = 0 ???$$

Arithmétique modulo 2

$+$	0	1
0		
1		

Arithmétique modulo 2

$+$	0	1
0	0	
1		

Arithmétique modulo 2

$+$	0	1
0	0	1
1		

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0		
1		

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	
1		

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1		

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

- On note $\mathbf{F}_2$ l'ensemble $\{0, 1\}$ muni de ces deux lois d'addition et de multiplication.

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

- On note $\mathbf{F}_2$ l'ensemble $\{0, 1\}$ muni de ces deux lois d'addition et de multiplication.
- Un élément de l'ensemble $\mathbf{F}_2$ s'appelle un *bit*.

Arithmétique modulo 2

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

- On note $\mathbf{F}_2$ l'ensemble $\{0, 1\}$ muni de ces deux lois d'addition et de multiplication.
- Un élément de l'ensemble $\mathbf{F}_2$ s'appelle un *bit*.
- Une suite de bits s'appelle un *mot* (binaire).

Un intérêt pratique

Les bits forment les briques du langage des ordinateurs :

Un intérêt pratique

Les bits forment les briques du langage des ordinateurs :
$0 \leftrightarrow$ le circuit est ouvert (le courant ne passe pas)

Un intérêt pratique

Les bits forment les briques du langage des ordinateurs :

$0 \leftrightarrow$ le circuit est ouvert (le courant ne passe pas)

$1 \leftrightarrow$ le circuit est fermé (le courant passe)

Un exemple d'encodage : ASCII

- On dispose de 128 caractères (caractères latins, chiffres, signes de ponctuation, etc) :

Un exemple d'encodage : ASCII

- On dispose de 128 caractères (caractères latins, chiffres, signes de ponctuation, etc) :

```
 !"#$%&'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]^_
`abcdefghijklmno
pqrstuvwxyz{|}~
```

Un exemple d'encodage : ASCII

- On dispose de 128 caractères (caractères latins, chiffres, signes de ponctuation, etc) :

```
 !"#$%&'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]^_
`abcdefghijklmno
pqrstuvwxyz{|}~
```

- Chaque caractère est remplacé par un mot de 7 bits.

Un exemple d'encodage : ASCII

- On dispose de 128 caractères (caractères latins, chiffres, signes de ponctuation, etc) :

```
 !"#$%&'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]^_
`abcdefghijklmno
pqrstuvwxyz{|}~
```

- Chaque caractère est remplacé par un mot de 7 bits.
- Par exemple, a est codé par 1100001, et 1111000 correspond à x.

On sait coder/décoder

Tout va bien !

On sait coder/décoder

Tout va bien ! Sauf si :

On sait coder/décoder

Tout va bien ! Sauf si :

On sait coder/décoder

Tout va bien ! Sauf si :

- le CD est rayé

On sait coder/décoder

Tout va bien ! Sauf si :

- le CD est rayé
- la connexion est de mauvaise qualité

On sait coder/décoder

Tout va bien ! Sauf si :

- le CD est rayé
- la connexion est de mauvaise qualité
- il y a du bruit, de l'interférence.

On sait coder/décoder

Tout va bien ! Sauf si :

- le CD est rayé
- la connexion est de mauvaise qualité
- il y a du bruit, de l'interférence.

Si on ne fait rien l'information (texte, voix, son, image) qui arrive à son destinataire est de mauvaise qualité, voire inintelligible.

Sommaire

Comment faire ?

Comment faire ?

Idée générale : il faut ajouter de la redondance

Comment faire ?

Idée générale : il faut ajouter de la redondance mais pas trop !

Code de parité : définition

- mot de 7 bits $\rightsquigarrow$ mot de 8 bits (=octet).

Code de parité : définition

- mot de 7 bits $\rightsquigarrow$ mot de 8 bits (=octet).
- Le 8-ième bit, appelé *bit de parité*, vaut

$$\begin{cases} 0 & \text{si le nombre de 1 du mot de 7 bits est pair} \\ 1 & \text{si le nombre de 1 du mot de 7 bits est impair} \end{cases}$$

Code de parité : définition

- mot de 7 bits $\rightsquigarrow$ mot de 8 bits (=octet).
- Le 8-ième bit, appelé *bit de parité*, vaut

$$\begin{cases} 0 & \text{si le nombre de 1 du mot de 7 bits est pair} \\ 1 & \text{si le nombre de 1 du mot de 7 bits est impair} \end{cases}$$

- Maintenant a est codé par 11000011 et x par 11110000

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

- 01101010 est-il un élément du code de parité ?

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

- 01101010 est-il un élément du code de parité ? OUI !

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

- 01101010 est-il un élément du code de parité ? OUI !
- 0000111 est un élément du code de parité.

Code de parité et addition

Définition

On appelle code de parité l'ensemble des mots de 8 bits de la forme

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 (b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7)$$

où $b_i = 0$ ou 1 et l'addition est celle de $\mathbf{F}_2$.

- 01101010 est-il un élément du code de parité ? OUI !
- 00001111 est un élément du code de parité.

Code de parité et détection

Code de parité et détection

- Si on change 1 bit dans un octet du code de parité, le résultat n'est plus dans le code de parité.

Code de parité et détection

- Si on change 1 bit dans un octet du code de parité, le résultat n'est plus dans le code de parité.
- Le code de parité détecte donc 1 erreur, mais il ne la corrige pas !

Code de parité et détection

- Si on change 1 bit dans un octet du code de parité, le résultat n'est plus dans le code de parité.
- Le code de parité détecte donc 1 erreur, mais il ne la corrige pas !
- On parle de *code détecteur*.

Le numéro INSEE

- Les deux derniers chiffres du numéro INSEE correspondent à la clé.

Le numéro INSEE

- Les deux derniers chiffres du numéro INSEE correspondent à la clé.

Le numéro INSEE

- Les deux derniers chiffres du numéro INSEE correspondent à la clé.

- La clé est définie par

$$97 - (\text{le reste de la division euclidienne de } N \text{ par } 97),$$

où N est le nombre formé des 13 premiers chiffres.

Les numéros ISBN

Pour le dernier chiffre - la clé - on calcule la somme S des douze premiers pondérée par les coefficients 1,3,1,3, etc.

- Si S est divisible par 10, la clé est 0.
- Sinon, c'est
  $10 -$ (le reste de la division euclidienne de S par 10)

Peut-on mieux faire ?

Comment corriger les erreurs trouvées ?

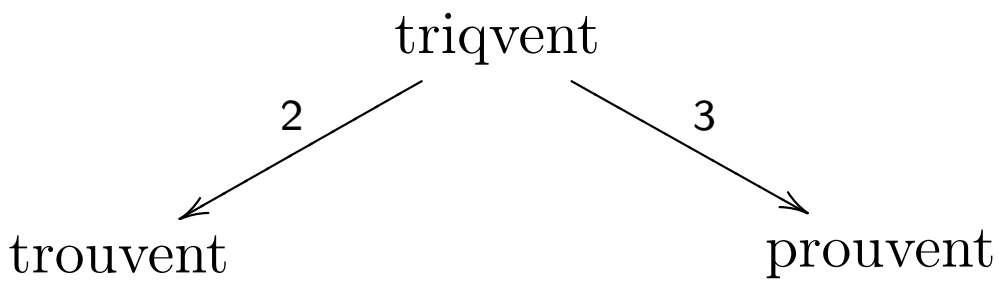
- Choisir un entier entre 0 et 15.

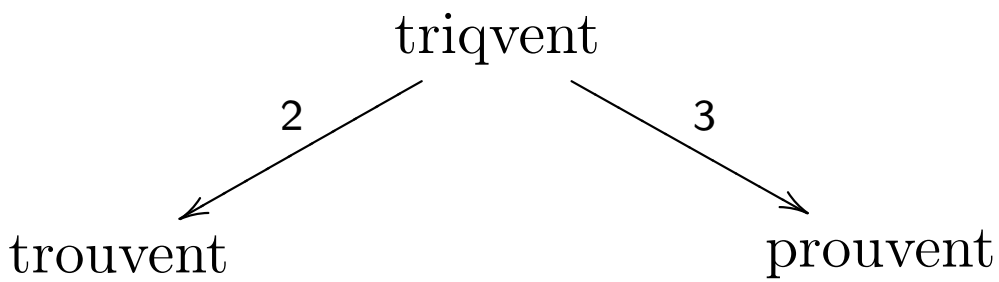
- Choisir un entier entre 0 et 15.
- Répondre aux sept questions suivantes. On a le droit de mentir (au plus) UNE fois. À la fin je révèle l'entier choisi (et l'éventuel mensonge !)

- Choisir un entier entre 0 et 15.
- Répondre aux sept questions suivantes. On a le droit de mentir (au plus) UNE fois. À la fin je révèle l'entier choisi (et l'éventuel mensonge !)

1. L'entier choisi est-il inférieur ou égal à 7 ?
2. L'entier choisi est-il dans l'ensemble $\{0, 1, 2, 3, 8, 9, 10, 11\}$?
3. L'entier choisi est-il dans l'ensemble $\{0, 1, 4, 5, 8, 9, 12, 13\}$?
4. L'entier choisi est-il pair ?
5. L'entier choisi est-il dans l'ensemble $\{0, 2, 5, 7, 9, 11, 12, 14\}$?
6. L'entier choisi est-il dans l'ensemble $\{0, 3, 4, 7, 9, 10, 13, 14\}$?
7. L'entier choisi est-il dans l'ensemble $\{0, 3, 5, 6, 8, 11, 13, 14\}$?

Sommaire


triqvent
2
3
trouvent
prouvent



On corrige au plus proche !

Une distance entre les mots binaires ?

Définition intuitive :

Une distance entre les mots binaires ?

Définition intuitive :

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\}$$

Une distance entre les mots binaires ?

Définition intuitive :

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\}$$

Par exemple, on a $d(1001001, 0001000) =$

Une distance entre les mots binaires ?

Définition intuitive :

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\}$$

Par exemple, on a $d(1001001, 0001000) = 2$.

Une distance entre les mots binaires ?

Définition intuitive :

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\}$$

Par exemple, on a $d(1001001, 0001000) = 2$.

Propriétés élémentaires

$$d(m, m) = 0 \quad \text{et} \quad d(m, m') = d(m', m)$$

quels que soient m et m'.

Distance et poids

Distance et poids

$$\omega(m) = \sharp\{1 \text{ apparaissant dans l'écriture du mot } m\}$$

s'appelle le *poids* du mot m.

Distance et poids

$$\omega(m) = \sharp\{1 \text{ apparaissant dans l'écriture du mot } m\}$$

s'appelle le *poids* du mot m.
Par exemple, on a

$$\omega(1001001) =$$

Distance et poids

$$\omega(m) = \sharp\{1 \text{ apparaissant dans l'écriture du mot } m\}$$

s'appelle le *poids* du mot m.
Par exemple, on a

$$\omega(1001001) = 3$$

Distance et poids

$$\omega(m) = \sharp\{1 \text{ apparaissant dans l'écriture du mot } m\}$$

s'appelle le *poids* du mot m.
Par exemple, on a

$$\omega(1001001) = 3 \quad \text{et} \quad \omega(0001000) =$$

Distance et poids

$$\omega(m) = \sharp\{1 \text{ apparaissant dans l'écriture du mot } m\}$$

s'appelle le *poids* du mot m.
Par exemple, on a

$$\omega(1001001) = 3 \quad \text{et} \quad \omega(0001000) = 1.$$

Poids et somme

Définition

On appelle somme des mots m et m' et on note $m + m'$ le mot (de même longueur que m et m') dont chaque bit est la somme des bits correspondants de m et m'.

Poids et somme

Définition

On appelle somme des mots m et m' et on note $m + m'$ le mot (de même longueur que m et m') dont chaque bit est la somme des bits correspondants de m et m'.

$$\begin{array}{cccccccc} & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ + & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline \end{array}$$

Poids et somme

Définition

On appelle somme des mots m et m' et on note $m + m'$ le mot (de même longueur que m et m') dont chaque bit est la somme des bits correspondants de m et m'.

$$\begin{array}{cccccccc} & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ + & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$

Poids et somme

Définition

On appelle somme des mots m et m' et on note $m + m'$ le mot (de même longueur que m et m') dont chaque bit est la somme des bits correspondants de m et m'.

$$\begin{array}{cccccccc} & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ + & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$

On a $\omega(m + m') \leq \omega(m) + \omega(m')$.

Distance, poids et somme

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\}$$

Distance, poids et somme

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\} = \omega(m + m')$$

Distance, poids et somme

$$d(m, m') = \sharp\{\text{bits différents entre } m \text{ et } m'\} = \omega(m + m')$$

D'où pour tous m, m', m'', on a

$$\begin{aligned} d(m, m') &= \omega(m + m'' + m'' + m') \\ &\leq \omega(m + m'') + \omega(m'' + m') \\ &= d(m, m'') + d(m'', m') \end{aligned}$$

L'inégalité triangulaire

$$d(m, m') \leq d(m, m'') + d(m'', m')$$

Distance de Hamming

Définition

La distance de Hamming d_H d'un code est la distance minimale entre 2 mots distincts du code.

Distance de Hamming

Définition

La distance de Hamming d_H d'un code est la distance minimale entre 2 mots distincts du code.

La distance de Hamming du code de parité est

Distance de Hamming

Définition

La distance de Hamming d_H d'un code est la distance minimale entre 2 mots distincts du code.

La distance de Hamming du code de parité est 2.

Distance de Hamming

Définition

La distance de Hamming d_H d'un code est la distance minimale entre 2 mots distincts du code.

La distance de Hamming du code de parité est 2.
[si je change 1 bit d'un élément du code de parité, le résultat n'est plus dans le code de parité, et inversement.]

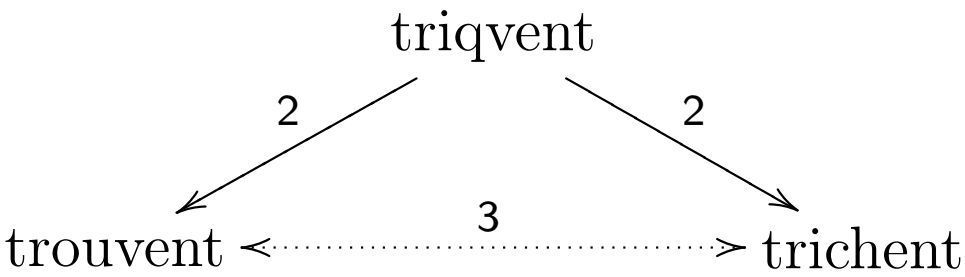
Détection et distance de Hamming

Détection et distance de Hamming

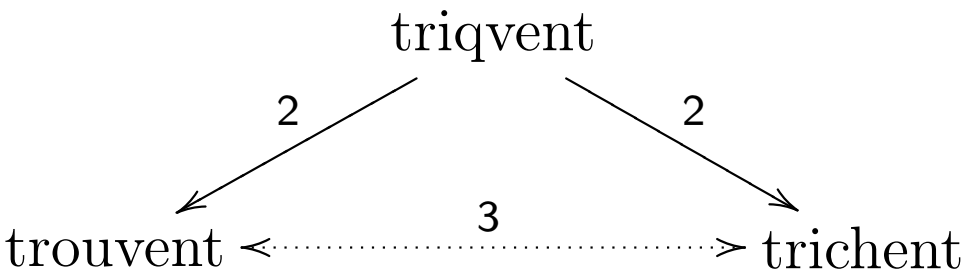
Un code de distance de Hamming $d_H = d$ détecte $d - 1$ erreurs.

Correction et distance de Hamming

Correction et distance de Hamming

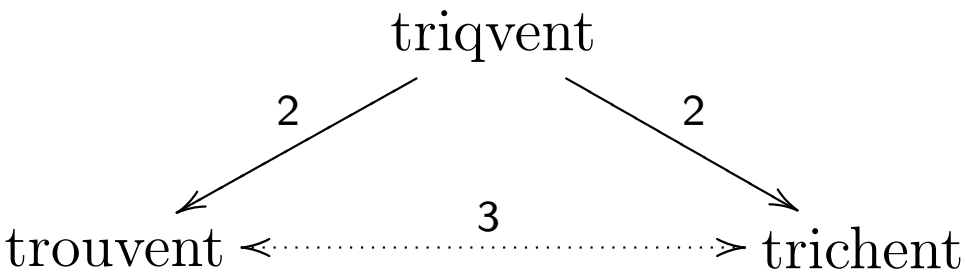


Correction et distance de Hamming



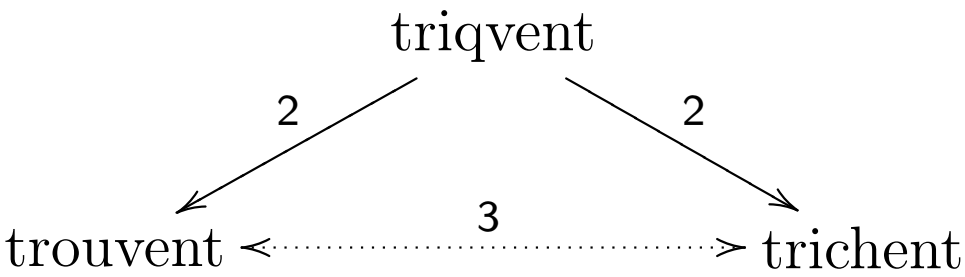
Lequel choisir ?

Correction et distance de Hamming



Lequel choisir ? On ne sait pas !

Correction et distance de Hamming



Lequel choisir ? On ne sait pas !

Un code de distance de Hamming d corrige $\lfloor \frac{d-1}{2} \rfloor$ erreurs.

En particulier...

Un code de distance 3 corrige $\lfloor \frac{3-1}{2} \rfloor = 1$ erreur...