

Les anagrammes de ce titre

Jean-Christophe Filliâtre
CNRS

Journées Francophones des Langages Applicatifs
30 janvier 2026

étant donnés

- un ensemble de mots
- un multiensemble de lettres

combien y a-t-il de concaténations de mots utilisant exactement toutes ces lettres ?

avec les lettres de

ancres

et les mots de notre correcteur orthographique, la réponse est 22 :

"rances"	"nacres"	"ecrans"	"cranes"	"carnes"	"ancres"
"se cran"	"es cran"	"cran se"	"cran es"	"nes car"	"nes arc"
"car nes"	"arc nes"	"cars ne"	"cars en"	"arcs ne"	"arcs en"
"ne cars"	"ne arcs"	"en cars"	"en arcs"		

- les accents sont ignorés
- la casse est ignorée
- un même mot peut être utilisé plusieurs fois
- l'ordre des mots importe

compter les anagrammes de « les anagrammes de ce titre », c'est-à-dire de

$$a_3 c d e_5 g i l m_2 n r_2 s_2 t_2$$

pour un dictionnaire de 133 949 mots

en un temps raisonnable

c'est un problème dans **#P-complet**

c'est-à-dire parmi les plus difficiles de la classe #P

#P = classe des fonctions qui comptent les solutions d'un problème de décision de NP

(preuve par réduction de #SAT)

pour avoir confiance dans le résultat, on se donne

- une batterie de tests
- une implémentation de référence (< 60 lignes)

entrée :

- un multiset M de lettres
- un dictionnaire D de mots

avant tout,

- ① on retire de D les mots non réalisables
⇒ plus que 7 994 mots
- ② on réunit les mots formant le même multiset de lettres
rances \simeq nacres \simeq ecrans \simeq cranes \simeq carnes \simeq ancres (compte $\times 6$)
⇒ plus que 5 394 classes

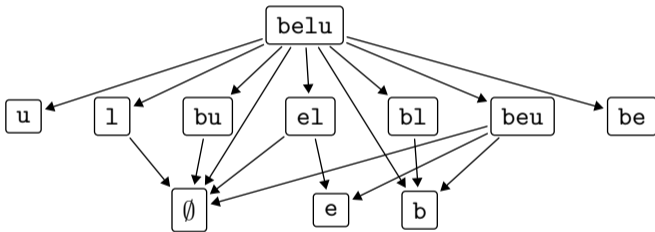
ensuite, on explore le graphe dont

- les sommets sont des multiensembles de lettres
- les arcs sont $(x) \xrightarrow{w} (y)$ avec w un mot, $w \subseteq x$ et $y = x \setminus w$

le nombre de chemins de M à \emptyset est la solution

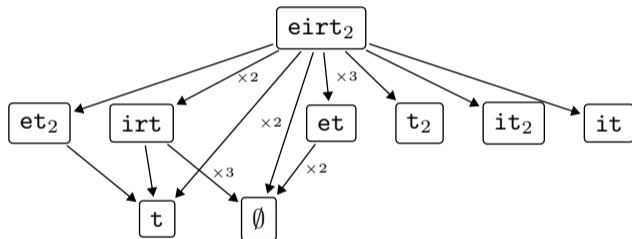
bien sûr, la mémorisation est notre amie

avec les lettres de « bleu »,



il y a 5 chemins

avec les lettres de « titre »,



il y a $2 \times 3 + 2 + 3 \times 2 = 14$ chemins

reste une question : comment représenter les multiensembles ?

en particulier, il faut

- tester l'inclusion
- calculer la différence
- parcourir tous les sous-multiensembles

```
module type MULTISSET = sig
  type elt
  type t
  val empty: t
  val occ: elt -> t -> int
  val add1: elt -> t -> t
  val remove1: elt -> t -> t
  val inclusion: t -> t -> bool
  val diff: t -> t -> t
  val iter_sub: (t -> t -> unit) -> t -> unit
  ...
end
```

si le multienemble n'est pas trop grand, il peut être représenté facilement par un entier

ainsi, tous les multiensembles contenus dans $a_3cde_5gilm_2nr_2s_2t_2$ peuvent être représentés sur 19 bits :

- 2 bits pour compter les a
- 1 bit pour compter les c
- 1 bit pour compter les d
- 3 bits pour compter les e
- etc.

par exemple comme ceci

18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
t	t	s	s	r	r	n	m	m	l	i	g	e	e	e	d	c	a	a

pour des capacités maximales annoncées, ici

a₃cde₅gilm₂nr₂s₂t₂

on pré-calculer une table comme ceci

	a	c	d	e	g	i	l	m	n	r	s	t
position	0	2	3	4	7	8	9	10	12	13	15	17
masque	3	1	1	7	1	1	1	3	1	3	3	3

- égalité, comparaison et hachage sont immédiats
- si $x \subseteq y$ alors $y - x$ est la différence multiensembliste
- si l'union $x \cup y$ est représentable, alors $x + y$ la calcule
- si $x \subseteq y$ alors $x \leq y$

pour un type d'éléments

```
module type UNIVERSE = sig
  type t
  val hash: t -> int
  val equal: t -> t -> bool
end
```

on fournit des multiensembles sur un univers borné

```
module Make(X: UNIVERSE) : sig
  val create: (X.t * int) list ->
    (module MULTISSET with type elt = X.t)
end
```

(avec possiblement Invalid_argument "capacity exceeded")

pour « les anagrammes de ce titre », c'est-à-dire

$a_3cde_5gilm_2nr_2s_2t_2$

on obtient

206 995 250 368 518 anagrammes

en explorant un graphe de 115 130 sommets et 27 946 451 arcs
en à peine plus d'une minute

deux heures de calcul pour

compter les anagrammes de ce titre

et au final le résultat

114 307 996 038 183 869 048

`https://usr.lmf.cnrs.fr/~jcf/anagrammes/`



et un paquet Opam mset est en route