

Partitions

Ensemble des parties

Dans ce problème, on veut calculer explicitement l'ensemble des parties d'un ensemble E . Cet ensemble peut être représenté par un tableau `numpy` ou une liste. Si $|E| = n$, alors $\mathcal{P}(E)$ est de cardinal 2^n . Notre problème est par nature de complexité exponentielle.

Programmation dynamique

Pour un élément x de E , si on sait construire toutes les parties de E qui ne contiennent pas x , alors on sait construire $\mathcal{P}(E)$. C'est la réunion de :

- l'ensemble \mathcal{E}_1 des parties de E qui ne contiennent pas x
- avec l'ensemble \mathcal{E}_2 obtenu en ajoutant x à chaque élément de \mathcal{E}_1 .

Il s'agit d'un algorithme de *programmation dynamique*.

Q1

Ecrire la fonction `ens_parties(E:list)->list[list]` qui prend en paramètre un ensemble E (sous forme de liste) et renvoie l'ensemble des parties de E (sous forme de liste). E peut très bien être un multi-ensemble. On tolère que E soit modifié au cours de l'algorithme.

```
[31]: E = [1,2,3]
      P=ens_parties(E)
      P, len(P)
```

```
[31]: ([[], [1], [2], [1, 2], [3], [1, 3], [2, 3], [1, 2, 3]], 8)
```

Ecriture binaire

Si $|E| = n$, il y a une bijection entre $\mathcal{P}(E)$ et $[0, 2^n - 1]$. Considérons $E = \{e_0, \dots, e_{n-1}\}$

Pour chaque entier $k \in [0, 2^n - 1]$, son code binaire tient sur n bits. Ce code est interprété comme suit : s'il y a un 0 en place i du code, alors l'élément e_i de E n'est pas dans le sous-ensemble associé à k , mais il y est si on trouve un 1 en position i .

Exemple avec $n = 4$ et $k = 11$: Alors $|\mathcal{P}(E)| = 16$, le code de k est 1011. Donc k est associé au sous-ensemble $\{e_0, e_2, e_3\}$.

Q2

Ecrire la fonction `toBin(x:int)->list[int]` qui calcule en binaire big-endian (avec bit de poids fort en 1ere position) l'écriture de x en binaire sur n bits sans zéro superflu.

Etablir la complexité de votre algorithme.

```
[33]: toBin(0), toBin(11), toBin(16)
```

```
[33]: ([0], [1, 0, 1, 1], [1, 0, 0, 0, 0])
```

Q3

Ecrire la fonction `binaire(n:int,x:int)->list[int]` qui calcule l'expression de x en binaire sur n bits. Il peut donc y avoir des zéros superflus.

```
[35]: binaire(4,1), binaire(4,11), binaire (7,11)
```

```
[35]: ([0, 0, 0, 1], [1, 0, 1, 1], [0, 0, 0, 1, 0, 1, 1])
```

Q4

Ecrire la fonction `ens_parties2(E)` qui prend en paramètre un ensemble E sous forme de liste et renvoie son ensemble des parties en passant par l'expression binaire des nombres.

```
[37]: ens_partie2(['a','b','c'])
```

```
[37]: ([], ['c'], ['b'], ['b', 'c'], ['a'], ['a', 'c'], ['a', 'b'], ['a', 'b', 'c'])
```