



## Génération des parties à 2 ou 3 éléments

Le but de ce programme est de retourner toutes les combinaisons possibles à 2 éléments ou à 3 éléments d'un ensemble fini. Pour cela, on rentrera en argument la liste des éléments de l'ensemble ainsi que le nombre d'éléments souhaités dans les parties de l'ensemble (donc 2 ou 3).

Python possède une commande « combinations » venant du module « itertools ». Cette commande permet de donner toutes les combinaisons possible de  $m$  éléments d'un ensemble fini à  $n$  éléments.

```
1 from itertools import combinations
2 print(list(combinations([1,2,3,5,7],2)))
```

```
>>> [(1,2), (1,3), (1,5), (1,7), (2,3), (2,5), (2,7), (3,5), (3,7), (5,7)]
```

Ici, le programme donne toutes les combinaisons possibles des parties à 2 éléments de l'ensemble  $E = \{ 1 ; 2 ; 3 ; 5 ; 7 \}$ . Toutes ? Non... il ne donne pas par exemple (2,1). Il faut donc compléter le programme comme suit :

```
1 from itertools import combinations
2 def generation_parties(L,m) :
3     if m>n :
4         return % pour vérifier que le nombre d'éléments des parties souhaité est bien inférieur ou égal au nombre
d'éléments de l'ensemble
5     elif m==2 :
6         return (a,b) for a in L for b in L if a!=b
7     elif m==3 :
8         return (a,b,c) for a in L for b in L for c in L if a!=b!=c
9     print(list(generation_parties([1,2,3,5, 7],2)))
```

```
>>> [(3,1), (3,7), (5,1), (5,7), (2,5), (1,3), (7,1), (5,3), (1,2), (2,1), (2,7), (1,5), (7,3), (3,2), (3,5), (5,2), (2,3),
(1,7), (7,2), (7,5)]
```

Ici, on souhaite faire des parties à 2 éléments d'un ensemble à 5 éléments. Le nombre de parties est donc de  $\binom{5}{2}$  soit 10.

Puisque on souhaite aussi faire apparaître les couples de la forme  $(a, b)$  et  $(b, a)$  on obtient donc 20 parties au total.