Combinatoire et dénombrement

Compétences attendues en fin de chapitre :

- Dans le cadre d'un problème de dénombrement, utiliser une représentation adaptée (ensembles, arbres, tableaux, diagrammes) et reconnaitre les objets à démontrer.
- Effectuer des dénombrements simples dans des situations issues de divers domaines scientifiques (informatique, génétique, théorie des jeux, probabilités).

Le mathématicien du chapitre :

James Stirling (1692-1770) est né en Écosse, près de la ville de Stirling. En 1730, Stirling publie son plus important traité mathématique, baptisé Méthodus différentialis. Il s'intéresse à des notions modernes telles la vitesse de convergence d'une suite. C'est dans cet ouvrage qu'il développe la formule qui porte son nom pour l'approximation de $n!$. Cette formule est cependant apparue quelques mois auparavant dans les travaux de Moivre mais Stirling l'a précisée ce qui fait sa renommée.

1) Principe additif et multiplicatif

On rappelle la notion vue en seconde : deux ensembles sont disjoints s'ils n'ont aucun éléments en commun.

a) Ensemble fini et cardinal

Définition :

Soit E un ensemble fini possédant n éléments.
Le cardinal de E, noté $Card(E)$, est le nombre d'éléments de E.

Remarques :

- Si $E = \emptyset$, alors $Card(E) = 0$
- Attention, certains ensembles ne sont pas finis. C'est le cas de $\mathbb{N}$ par exemple.
- La notion de cardinal s'apparente à la notion d'effectif en statistiques.

Exemple :

Les jours de la semaine forment un ensemble fini. Ainsi, on a, avec les notations précédentes :
$E = \{Lundi; Mardi; Mercredi; Jeudi; Vendredi; Samedi; Dimanche\}$ et $Card(E) = 7$

b) Le principe additif

Propriété :

Soit p un entier naturel supérieur ou égal à 2.
Soient $A_1, A_2, A_3, \dots \dots, A_p$ p ensemble deux à deux disjoints. On a :
$Card\left(A_1 \cup A_2 \cup A_3 \dots \dots \cup A_p\right) = Card(A_1) + Card(A_2) + \dots\dots + Card\left(A_p\right)$

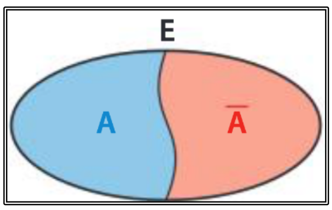
Attention :

Cette formule est fausse si les ensembles ne sont pas deux à deux disjoints. En effets, des éléments seraient comptés plusieurs fois dans différents ensembles. C'est la même notion que la formule de Poincaré vue en probabilité.

Cas particulier :

Soit A une partie d'un ensemble fini E. On note $\bar{A}$ son complémentaire. Alors A et $\bar{A}$ forment une partition de E. On a donc :

$$Card(\bar{A}) = Card(E) - Card(A)$$



c) *Le principe multiplicatif*

René Descartes a œuvré dans de nombreux domaines mathématiques. Le principe multiplicatif, aussi appelé produit cartésien lui est du.

Propriété :

Soient E et F deux ensembles non vides.
Le produit cartésien de E par F est : $E \times F = \{(x; y), x \in E, y \in F\}$

Remarque :

Il est important de noter que $E \times F \neq F \times E$ si E et F sont différents. Il suffit de penser aux coordonnées d'un point dans un repère cartésien.

Exemple :

On donne $E = \{1; 2; 3\}$ et $F = \{5; 6\}$

```python
import itertools
def produit(L,P):
    for i in itertools.product(L,P):
        print (i)
```

```
› produit([1,2,3],[5,6])
(1, 5)
(1, 6)
(2, 5)
(2, 6)
(3, 5)
(3, 6)
```

Propriété :

Soient E et F deux ensembles finis, alors : $Card(E \times F) = Card(E) \times Card(F)$

Exemple :

Dans l'exemple ci-dessus, on a : $Card(E) = 3$ et $Card(F) = 2$. On obtient 6 éléments.
Les éléments des listes ne sont pas nécessairement des nombres.

Remarque :

Attention, le symbole $\times$ ne représente pas le même objet. Dans $Card(E \times F)$, il représente le produit cartésien de deux ensembles tandis que dans $Card(E) \times Card(F)$, il représente la multiplication de deux entiers naturels.

d) *Les p-uplets*

Définition :

Soit p un entier supérieur ou égal à 2 et $E_1, E_2, E_3, \dots \dots, E_p$ p ensemble non vides
Un p-uplet (ou p-liste) est une liste ordonnée $(x_1, x_2, \dots, x_p)$ avec $x_i \in E_i$ pour $i \in [|1; p|]$

Remarque :

L'ensemble de ces p-uplets est donc le produit cartésien $E_1 \times E_2 \times \dots . \times E_p$

Vocabulaire :

Un 2-uplet est appelé un couple, un 3-uplet un triplet.

2) *p-uplets d'un ensemble fini*

a) *Nombre de p-uplets d'un ensemble à n éléments*

Activité introductive :

Une urne contient cinq boules notées $\{a, b, c, d, e\}$. On tire au hasard 3 boules de l'urne.
Les boules sont tirées ***une à une*** successivement de l'urne, ***avec remise*** de chaque boule tirée dans l'urne.
Ainsi, chaque boule a autant de chance d'être tirées et peut apparaître plusieurs fois.
Par exemple, si on a tiré b puis e puis b, un tel tirage sera noté (b, e, b).
La notation entre parenthèses signifie que l'on prend en compte l'ordre des boules tirées.
Ainsi $(b, e, b) \neq (e, b, b)$.
Un tel tirage sera appelé ***une liste à trois éléments*** de l'ensemble $\{a, b, c, d, e\}$.

1. Dénombrer tous les tirages possibles (on pourra s'aider d'un arbre).
2. Dénombrer tous les tirages tels que la 2ème boule tirée ne soit pas b.
3. Dénombrer tous les tirages tels que la boule B_3 soit tirée exactement deux fois.

Correction :

1. La construction d'un arbre n'est pas la meilleure idée. En effet, il y a 5 possibilités pour chaque boule soit au total $5^3 = 125$ soit un arbre avec 125 chemins…
2. Il est plus simple de dénombrer le nombre de tirage avec b en deuxième position. Il y en a 25. Il y a donc 100 tirages pour lesquels b n'est pas en 2ème position.
3. Il y a 4 possibilités pour que c soit tirée exactement 2 fois. Il faut ensuite dénombrer la place des boules c. On doit placer 2 boules c sur trois emplacements soit 3 possibilités. Il y a donc 12 tirages pour que c soit tirée exactement deux fois.

Définition :

Soit p un entier naturel non nul et E un ensemble non vide.
Un p-uplet (ou p-liste) d'éléments de E est un élément du produit cartésien E^p

Exemple :

Soit E l'ensemble des 26 lettres de l'alphabet. $(t, e, r, m, i, n, a, l, e)$ est un 9-uplet de E^9.

Générer un p-uplet avec python

```python
from random import randint
def p_uplet(p,L):
    n = len(L)
    p_uplet = []
    for i in range(1,p+1):
        x = randint(0,n-1)
        p_uplet.append(L[x])
    return p_uplet
```

```
> p_uplet(2,[5,6,7,8,9])
[9, 9]
> p_uplet(5,[1,2,3,4])
[3, 1, 1, 3, 3]
> p_uplet(3,["a","b","c","d","e"])
['b', 'c', 'c']
> p_uplet(2,[5,6,7,8,9])
[5, 9]
```

b) ***Permutations***

On reprend notre exemple de boules. Les boules sont tirées ***une à une*** successivement de l'urne, ***sans remettre*** dans l'urne les boules déjà tirées. Il y a donc équiprobabilité d'apparition, mais pas de répétition d'une même boule dans le tirage.

Ainsi, si on a tiré c puis a puis d. Un tel tirage sera encore noté à l'aide de parenthèses : (c, a, d) puisque l'ordre a toujours une importance.
1. Dénombrer tous les tirages possibles.
2. Dénombrer tous les tirages pour lesquels la 2ème boule tirée ne soit pas b.
3. Dénombrer tous les tirages ordonnés (lettre dans l'ordre alphabétique)

Correction :

1. On a 5 chances pour la première boule puis 4 pour la seconde et enfin 3 pour la dernière soit un total de 60 tirages possibles.
2. Il est plus simple de dénombrer les tirages où b est en deuxième position. Il y a 4 chances pour la première boule (puisqu'il ne peut y avoir b), 1 seule pour la seconde et 3 pour la dernière soit en multipliant 12 tirages. Il y a donc 48 tirages où b n'est pas en deuxième position.
3. La première boule est soit a, soit b, soit c. Si c'est a, on a 6 tirages possibles. Si c'est b, on a 3 tirages possibles et si c'est c, on a 1 seul tirage soit un total de 10 tirages pour avoir les indices croissants.

Définition :

Soit n un entier naturel non nul et $p \in [|1; n|]$. Soit E un ensemble fini de cardinal n,
Le nombre de p-uplet d'éléments deux à deux distincts de E est donné par :
$n(n-1)(n-2) \dots (n-p+1)$ avec p facteurs

Exemple :

Soit E l'ensemble des 26 lettres de l'alphabet. Je souhaite former des mots de 4 lettres avec toutes les lettres différentes, même s'ils n'ont pas de sens. Il y en a au total 358800

Définition :

On appelle permutation d'une ensemble E fini à n éléments tout n-uplet d'éléments deux à deux distincts de E.

Générer une permutation avec python

```python
from random import randint
def permutation(L):
    permutation = []
    rang = len(L)
    for i in range(rang):
        x = randint(0,rang-1)
        permutation.append(L[x])
        del L[x]
        rang = rang - 1
    return permutation
```

```
> permutation([1,2,3,4])
[2, 3, 4, 1]
> permutation([3,6,9,12,15])
[9, 12, 15, 6, 3]
> permutation(["a","b","c","d","e"])
['b', 'd', 'e', 'a', 'c']
```

On génère ainsi une liste d'éléments de E uniques et pas dans le même ordre.

c) Factorielle

Définition :

On appelle factorielle n le nombre entier $n! = n \times (n-1)(n-2) \times \ldots \times 2 \times 1$.

Exemples :

```python
def factorielle(n):
    f = 1
    for i in range(1,n+1):
        f = f*i
    return f
```

Par convention, : $0! = 1$. Puis on a :

```
> factorielle(2)
2
> factorielle(3)
6
> factorielle(4)
24
```

Remarques :

- Afin de maitriser le calcul, les premières factorielles sont à connaître par cœur.
- On obtient les autres à l'aide de la TI-83. Pour cela, à l'aide de la touche tests A math

```
NORMAL FLOTT AUTO RÉEL RAD MP
MATH NBRE CMPLX PROB FRAC
1:NbrAléat
2:Arrangement
3:Combinaison
4:!
```

```
NORMAL FLOTT AUTO RÉEL RAD MP
5!
                                   120
```

Propriétés :

Le nombre de permutation d'un ensemble à n éléments vaut $n!$

La formule de Stirling :

Peu de temps après Moivre, Stirling a montré que $n!$ pouvait être approché par le nombre : $n! \approx \sqrt{2\pi n}n^n e^{-n}$. Voici quelques exemples numériques qui montrent que plus n est grand, plus la formule de Stirling approche $n!$

$n = 4$	$n! = 24$	$n! \approx 23{,}51$
$n = 8$	$n! = 40320$	$n! \approx 39902$
$n = 15$	$n! = 1307674368000$	$n! \approx 1{,}300431 \times 10^{12}$

La notion d'équivalent sera enseignée en MPSI.

3) ***Parties d'un ensemble et combinaison***

a) ***Nombre de parties d'un ensemble***

Définition :

Soit E un ensemble. Dire qu'un ensemble F est une partie de E (ou que F est un sous ensemble de E, ou que F est inclus dans E) signifie que tous les éléments de F sont dans E.

Remarque :

Il ne faut pas confondre un p-uplet avec une partie. $(1; 2) = (2; 1)$ est une partie à 2 éléments de $\mathbb{N}$ tandis que les deux 2-uplets $(1; 2)$ et $(2; 1)$ sont différents

Propriété :

Soit E un ensemble fini à n éléments. Le nombre de parties de E vaut 2^n.

Démonstration :

Soit $E = \{e_1; e_2; e_3; \dots; e_n\}$ un ensemble fini à n éléments. On associe à chaque partie P de E un n-uplet de l'ensemble $\{0; 1\}$ de la manière suivante : $\forall i \in [|1; n|]$, on note 1 si e_i est dans P et 0 sinon. On écrit en fait un code binaire. Par exemple on associe à $\{e_1; e_3\}$ le n-uplet $(1; 0; 1; 0; \dots; 0)$. Ainsi le nombre de partie de E est égale au nombre de n-uplet de $\{0; 1\}$ soit donc avec le produit cartésien 2^n.

b) ***Combinaisons***

Activité introductive :

Les trois boules sont tirées simultanément (en une seule fois) de l'urne.
On ne peut pas parler ***d'ordre d'apparition*** des boules dans le tirage puisqu'elles sont tirées simultanément. Ainsi, une boule ne peut apparaître ***qu'une seule fois*** par tirage
On a tiré a, b, d. Un tel tirage sera noté $\{a, b, d\}$, la notation entre accolades signifiant qu'il n'y a pas d'ordre d'apparition et que les éléments du tirage sont deux à deux distincts.
Autrement dit, un tirage simultané de trois boules n'est autre qu'un sous-ensemble (ou partie) à trois éléments de l'ensemble $\{a, b, c, d, e\}$.
Ainsi $\{a, b, d\} = \{a, d, b\} = \{b, a, d\} = \{b, d, a\} = \{d, a, b\} = \{d, b, a\}$
Il est légitime de se demander combien il y a de tirages possibles. Pour cela, posons 5 stylos sur la table et fabriquons des ensembles de 3 stylos. On en dénombre 10.

Remarques :

- Il est très pertinent de remarquer que prendre 3 éléments parmi 5 revient à en laisser 2, ce qui est plus simple à compter.
- De plus, il y a 6 fois moins de tirages possibles lorsque le tirage est simultané que lorsqu'il est successif et sans remise. Cette différence s'explique puisque l'ordre n'est pas pris en compte. Les tirages $\{a, b, d\}$ et $\{d, b, a\}$ ne sont comptés que pour un.

Propriété :

Soient n et p deux entiers naturels avec $0 \leq p \leq n$ et E un ensemble de cardinal n.
On appelle combinaison de p éléments de E toute partie de E à p éléments.

Exemple :

On souhaite former un couple de délégués dans une classe de 30 élèves. Il y a 30 possibilités pour le premier élève puis 29 pour le second. Chaque couple (non ordonné) apparait deux fois. On a : $\frac{30\times29}{2} = 435$ couples possibles. C'est le nombre de parties à 2 éléments parmi 30.
Ce calcul ne prend pas en compte la parité nécessaire dans les élections de nos jours.

Propriété :

Soient n et p deux entiers naturels avec $0 \leq p \leq n$. Le nombre de combinaison de p éléments parmi n est donné par : $\binom{n}{p} = \frac{n(n-1)(n-2)\ldots(n-p+1)}{p!} = \frac{n!}{p!(n-p)!}$

Un tel nombre est appelé coefficient binomial.

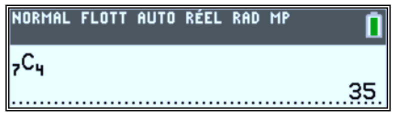
Remarques :

- $\binom{n}{p}$ se lit p parmi n. Vos parents ont appris la notation C_n^p
- On doit connaitre les premières valeurs des combinaisons pour se faciliter la vie.

Avec la TI-83 :

A la calculatrice, pour calculer $\binom{7}{4}$, on tape:

$\binom{7}{4} = \frac{7 \times 6 \times 5 \times 4}{24} = 35$



Dans un groupe de 7 personnes, il y a donc 35 quadruplés possibles différents.

Exercice :

Un sac contient 10 boules blanches et 7 boules noires. On tire simultanément 2 boules du sac.

1. Combien y-a-t-il de tirages possibles ?
2. Combien y-a-t-il de façons de tirer :

a. Deux boules blanches ? b. Deux boules noires ? c. Deux boules de même couleur ?
d. Au moins une boule noire ?

Correction :

1. On cherche une combinaison de 2 boules parmi 17 soit : $\binom{17}{2} = \frac{17\times16}{2} = 136$
2. Deux boules blanches : $\binom{10}{2} = \frac{10\times9}{2} = 45$

   Deux boules noires : $\binom{7}{2} = \frac{7\times6}{2} = 21$

   Deux boules de même couleur : 66 façons.

   Au moins une boule noire :$136 - 45 = 91$ façons.

Problème ouvert :

A-t-on plus de chances de gagner au Loto ou à l'Euro millions ?

4) Propriétés des combinaisons

a) Propriétés

On peut trouver les premiers coefficients binomiaux avec un peu de bon sens.

- Dans un ensemble à n éléments, combien y-a-t-il de manière de prendre aucun élément ? Une seule manière, je les laisse tous donc $\binom{n}{0} = 1$
- Dans un ensemble à n éléments, combien y-a-t-il de manière de prendre un élément ? Il y a n manières, je les prends les uns après les autres donc $\binom{n}{1} = n$
- Dans un ensemble à n éléments, combien y-a-t-il de manière de prendre deux éléments ? C'est le même exemple que pour les délégués donc $\binom{n}{2} = \frac{n(n-1)}{2}$

Propriété :

Soient n et p deux entiers naturels avec $0 \leq p \leq n$. Alors $\binom{n}{p} = \binom{n}{n-p}$

Remarque :

Dénombrer le nombre de parties à p éléments revient à dénombrer les parties à $n - p$ éléments. Les coefficients binomiaux sont symétriques. On a donc :

$\binom{n}{0} = 1 = \binom{n}{n}, \binom{n}{1} = n = \binom{n}{n-1}$ et $\binom{n}{2} = \frac{n(n-1)}{2} = \binom{n}{n-2}$

Propriété :

Pour tout entier naturel n, on a : $\sum_{p=0}^{n} \binom{n}{p} = 2^n$

Démonstration :

$\binom{n}{p}$ représente le nombre de parties à p éléments de E. En ajoutant toutes les parties de E en fonction de leur nombre d'éléments, on obtient le résultat obtenu plus haut.

b) *Le triangle de Pascal*

Propriété :

Soient n et p deux entiers naturels avec $0 \leq p \leq n$. Alors on a: $\binom{n}{p} = \binom{n-1}{p-1} + \binom{n-1}{p}$

Démonstration :

On pourrait démontrer cela avec la loi binomiale mais nous l'apprendrons plus tard.
La démonstration est technique et joue sur le calcul de factorielles.

Remarques :

- Cette propriété permet donc de calculer de proche en proche les coefficients.
- Et d'obtenir ainsi le triangle de Pascal.
- La généralisation des identités remarquables du collège, appelé binôme de Newton s'obtient à l'aide du triangle de Pascal. Il sera enseigné en maths expertes.

Générer le triangle de pascal en python

```python
def pascal(n):
  T=[[1],[1,1]]
  print (T[0])
  print (T[1])
  for i in range(2,n+1):
    T.append([1])
    for j in range(1,i):
      T[i].append(T[i-1][j-1]+T[i-1][j])
    T[i].append(1)
    print(T[i])
```

```
> pascal(5)
[1]
[1, 1]
[1, 2, 1]
[1, 3, 3, 1]
[1, 4, 6, 4, 1]
[1, 5, 10, 10, 5, 1]
```

Pour les élèves qui n'ont pas la chance d'être en math experte, voici à quoi ressemble le binôme de Newton :
$(a+b)^5 = 1a^5 + 5a^4b + 10a^3b^2 + 10a^2b^3 + 5ab^4 + 1b^5$

Propriété :

Soient a et b deux nombres réels quelconques. On a la formule du binôme de Newton :

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k\, b^{n-k}$$

Remarque :

En remplaçant a et b par 1, on retrouve la propriété du haut de la page.