Ensembles de nombres de somme donnée

Un nombre entier N étant donné, on considère la suite des N^2 nombres entiers de 0 à $N^2 - 1$. Remarquons que la somme de ces nombres est $N^2(N^2 - 1)/2$ .A partir de là, on veut obtenir tous les sous-ensembles formés de N éléments de cette suite, dont la somme vaut $S = N (N^2 - 1) / 2$. Par exemple pour $N = 4$, avec les nombres de 0 à 15, on veut avoir tous les ensembles formés de 4 de ces nombres, avec une somme égale à 30, comme 0 1 14 15, 0 2 13 15, 0 3 12 15, 0 3 13 14, 0 4 11 15, 0 4 12 14, 0 5 10 15, 0 5 11 14, 0 5 12 13, 0 6 9 15, …, 5 6 7 12, 5 6 8 11, 5 6 9 10, 5 7 8 10, 6 7 8 9. Ici nous avons écrit chacun de ces mots dans l'ordre croissant de ses lettres (nombres), et la succession des mots est aussi dans l'ordre croissant. Autrement dit, nous avons choisi un ordre là où il n'y avait pas d'ordre.

1) Vérifier que le dernier mot est toujours formé de N nombres successifs dont le premier est $N(N - 1) /2$, comme 6 7 8 9 pour $N = 4$

En effet, la somme de ces N nombres à partir de $N(N - 1) /2$, est égale à :

$$N\, N(N - 1) /2 + 1 + 2 + 3 + \ldots + N - 1 = N\, N(N - 1) /2 + N(N - 1) /2$$
$$= N(N - 1) (N + 1) /2 = S.$$

Il s'agit bien d'une solution du problème, et comme les N nombres se succèdent, on ne peut pas trouver de mot qui le suive, c'est le dernier.

2) Faire le programme pour $N = 3$

Il suffit de faire une triple boucle lâchant trois nombres dans l'ordre croissant, et de garder ceux dont la somme vaut S.

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#define N 3
#define NN  (N*N)
int compteur,S;

int main()
{  int i,j,k;
   S=(NN-1)*N/2;
   for(i=0;i<=N*(N-1)/2;i++) for(j=i+1;j<NN;j++) for(k=j+1;k<NN;k++)
   if (i+j+k==S) {compteur++; printf("\n%3.d : (%d %d %d) ",compteur,i,j,k);}
   return 0;
}
```

On trouve huit solutions :

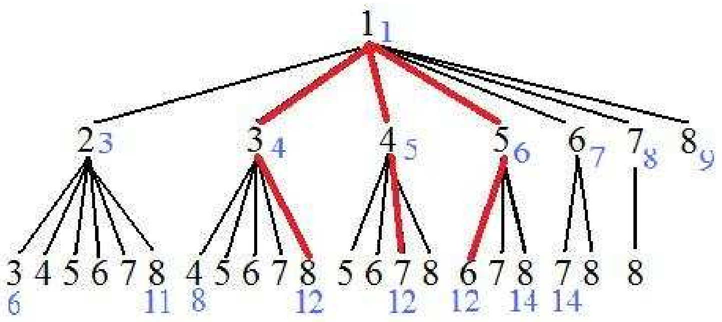
```
1 : (0 4 8)
2 : (0 5 7)
3 : (1 3 8)
4 : (1 4 7)
5 : (1 5 6)
6 : (2 3 7)
7 : (2 4 6)
8 : (3 4 5)
```

Mais un tel programme ne marche que pour $N = 3$, avec trois boucles imbriquées. Pour d'autres valeurs de N, le nombre de boucles doit changer, et il faut réécrire le programme.

3) Faire un programme qui marche pour une valeur quelconque de *N*

On va fabriquer des arbres de racines 0, 1, 2, … , $N(N\text{-}1)/2$ qui sont les premiers éléments possibles de chaque ensemble cherché. La fonction *arbre*() se rappelle à partir de cette racine i, soit *arbre*(i,0,i) avec l'étage 0 et le poids i. L'arbre construit va posséder les étages 0, 1, …, $N - 1$. Si l'on n'est pas encore à l'étage $N - 1$ et que l'on n'a pas atteint par cumul le poids S, la fonction *arbre*(i, *etage*, *poids*) se rappelle sur ses successeurs possibles, de $i + 1$ à $N^2 - 1$. Lorsque l'on est à l'étage $N - 1$ et que le poids S est obtenu, on fait l'affichage de la branche concernée à partir de la racine.

Le dessin suivant donne l'arbre obtenu pour $N = 3$ à partir de l'élément 1 comme racine, avec en rouge les branches donnant les solutions, et en bleu quelques poids obtenus par cumul



Programme :

```
int main()
{  int i;
   S=(NN-1)*N/2;
   for(i=0;i<=N*/(N-1)/2;i++) arbre(i,0,i);
   printf(" ******  %d  ******", compteur);
   return 0;
}

void arbre( int i,int etage, int poids)
  {   int k;
      if (etage==N-1 && poids==S)
         { compteur++;
           printf("%d : ",compteur);
           for (k=1;k<=etage;k++) printf("%d ", predecesseur[k]);
           printf("%d \n",i);
         }
      else if (poids<S && etage<N-1)
      for(k=i+1;k<NN;k++)
         { predecesseur[etage+1]=i;
           arbre(k,etage+1,poids+k);
         }
  }
```

On trouve les résultats suivants :

N	*Nombre de solutions*
2	2
3	8
4	86
5	1394
6	32 134
7	957 332
8	35 154 340

4) Quelques indications théoriques

- Prenons l'exemple de $N = 3$. On veut trouver, de toutes les façons possibles, trois nombres entiers x, y, z tels que $x + y + z = 12$, avec $0 \leq x < y < z \leq 8$. Procédons au changement d'inconnues (de façon à ne plus avoir la contrainte d'ordre croissant pour x, y, z):

$X = x$, $Y = y - x$, $Z = z - y$ ou inversement $x = X$, $y = X + Y$, $z = X + Y + Z$. Les contraintes deviennent :

$3X + 2Y + Z = 12$ avec $X \geq 0$, $Y, Z > 0$ et $X + Y + Z \leq 8$

Pour les contraintes : $3X + 2Y + Z = 12$ avec $X \geq 0$, Y, $Z > 0$, on a la fonction génératrice associée :
$(1+ q^3 + q^6 + q^9 + \ldots)(q^2 + q^4 + q^6 + \ldots)(q + q^2 + q^3 + \ldots)$. Le coefficient du terme en q^{12} donne le nombre de solutions associées, soit ici 12. [1]

Il reste la contrainte $X + Y + Z \leq 8$. Parmi les solutions précédentes, il convient d'enlever toutes les solutions telles que :

$X + Y + Z > 8$. En tenant compte du fait que $Z = 12 - 3X - 2Y$, la contrainte devient $2X + Y < 4$, ou encore $2X + Y \leq 3$, On prend donc tous les X, Y vérifiant cette contrainte, mais sous réserve que le Z qui leur correspond, soit $Z = 12 - 3X - 2Y$, soit strictement positif. Pour les (X, Y) vérifiant $2X + Y \leq 3$, il leur correspond la fonction génératrice :
$(1 + Z^2 + Z^4 + \ldots)(Z + Z^2 + Z^3 + \ldots)$, la somme des termes de degré inférieur ou égal à 3 donnant le nombre de solutions.

On trouve les termes : $1Z$, $1Z^2$, $1Z^3$ et $Z^2\, Z$, soit 4 solutions. Les valeurs de (X, Y) correspondantes sont (0, 1), (0, 2), (0 3), (1, 1). Les valeurs respectives de Z sont 10, 8, 6, 7, toutes positives. Ces quatre valeurs conviennent.

[1] Pour trouver cette valeur, il suffit d'utiliser un logiciel mathématique. Ainsi, avec *Mathematica*, on appelle :

Series[$\frac{1}{1-q^3}\frac{q^2}{1-q^2}\frac{q}{1-q}$,{q,0,12}] et l'on prend le coefficient du terme en q^{12} dans le développement en série de la fonction

Le nombre de solutions du problème est donc 12 – 4 = 8.

- Pour $N = 4$, on a les contraintes
$4X + 3Y + 2Z + T = 30$ avec $X \geq 0$, Y, Z et $T > 0$ et $X + Y + Z + T \leq 15$.

On prend la fonction génératrice associée à $4X + 3Y + 2Z + T = 30$ avec $X \geq 0$, Y, Z et $T > 0$, soit $\frac{1}{1-q^4}\frac{q^3}{1-q^3}\frac{q^2}{1-q^2}\frac{q}{1-q}$. Le coefficient du terme en Z^{30} est 169. Mais on doit enlever à ces solutions celles vérifiant $X + Y + Z + T > 15$. Cela revient à enlever toutes les solutions de $3X + 2Y + Z \leq 14$, sous réserve qu'alors $T = 30 - 4X - 3Y - 2Z$ soit strictement positif. Pour $3X + 2Y + Z \leq 14$, on a la fonction génératrice associée $\frac{1}{1-q^3}\frac{q^2}{1-q^2}\frac{q}{1-q}$ dont on prend la somme des coefficients des termes de degré inférieur ou égal à 14, ce qui donne 83 cas. On vérifie que tous ces cas sont tels que $T > 0$. Par exemple pour $3X + 2Y + Z = 14$, on a 16 cas avec les quadruplets (X, Y, Z, T) qui sont : (0 1 12 3)(0 2 10 4)(0 3 8 5) …(0 6 2 8), (1 1 9 5)(1 2 7 6) …(1 5 1 9), (2 1 6 7)(2 2 4 8)(2 3 2 9), (3 1 3 9)(3 2 1 10). On fait de même avec $3X + 2Y + Z = 13$ et ses 15 cas avec les valeurs de T qui démarrent plus haut que précédemment. Et ainsi de suite jusqu'à $3X + 2Y + Z = 3$ et son cas unique.

Finalement le nombre de solutions est 169 – 83 = 86.

- Pour $N = 5$, on a le système :

$5X + 4Y + 3Z + 2T + U = 60$ avec $X \geq 0$, Y, Z, T, $U > 0$, et $X + Y + Z + T + U \leq 24$.

Les contraintes $5X + 4Y + 3Z + 2T + U = 60$ avec $X \geq 0$, Y, Z, T, $U > 0$ donnent la fonction génératrice $\frac{q^5}{1-q^5}\frac{q^4}{1-q^4}\frac{q^3}{1-q^3}\frac{q^2}{1-q^2}\frac{q}{1-q}$ dont le terme en q^{60} a pour coefficient 3765. Mais il faut enlever à ce nombre de solutions toutes celles telles que $X + Y + Z + T + U \leq 24$. Avec $U = 60 - 5X - 4Y - 3Z - 2T$, cela revient à chercher les solutions de $4X + 3Y + 2Z + T \leq 35$ sous réserve que le U qui leur est associé soit > 0. Le nombre de solutions de $4X + 3Y + 2Z + T \leq 35$ s'obtient en sommant les coefficients des termes de degré ≤ 35 du développement de $\frac{1}{1-q^4}\frac{q^3}{1-q^3}\frac{q^2}{1-q^2}\frac{q}{1-q}$ ce qui donne 2427 solutions. Mais on remarque que certaines, obtenues pour les plus hauts degrés, ne donnent pas $T > 0$. Il convient de les retrancher du résultat précédent.

Ainsi, pour $4X + 3Y + 2Z + T = 35$, on trouve 270 solutions. Mais le fait d'avoir $U = 25 - (X + Y + Z + T)$ impose que $X + Y + Z + T < 25$ ce qui n'est pas toujours le cas. Il suffit de prendre les premières solutions de $4X + 3Y + 2Z + T = 35$ pour le constater. On a les quadruplets (X, Y, Z, T) : (0 1 1 30), (0 1 2 28),…, (0 1 8 16), (0 2 1 27), …(0 2 6 17), (0 3 1 24), …, (0 3 4 18), (0 4 1 21), (0 4 2 19), soit 20 cas commençant par 0, puis 9 cas commençant par 1 et 2 cas commençant par 2, soit un total de 31. Le nombre de solutions est 270 – 31 = 239.

Cela peut être retrouvé directement. Parmi les 270 solutions il faut seulement garder celles vérifiant $X + Y + Z + T < 25$. Avec $T = 35 - 4X - 3Y - 2Z$, cela impose que $3X + 2Y + Z > 10$. Aux 270 solutions il convient d'enlever toutes les solutions de $4X + 3Y + 2Z \leq 10$, sous réserve que le T associé soit >0. Le développement de la fonction génératrice associée à $4X + 3Y + 2Z \leq 10$ donne justement 31 cas, tous ayant leur $T > 0$.

Ce que l'on a fait pour $4X + 3Y + 2Z + T = 35$, avec la contrainte supplémentaire $4X + 3Y + 2Z \leq 10$, on le fait à son tour pour $4X + 3Y + 2Z + T = 34$, avec la contrainte supplémentaire $4X + 3Y + 2Z \leq 8$, soit 249 – 16 = 233 cas, puis pour $4X + 3Y + 2Z + T = 33$, avec la contrainte supplémentaire $4X + 3Y + 2Z \leq 6$, avec 225 – 7 = 218 solutions, puis pour $4X + 3Y + 2Z + T = 32$, avec la contrainte supplémentaire $4X + 3Y + 2Z \leq 4$, avec 206 – 2 = 204 solutions. Enfin, pour toutes les valeurs inférieures à 32, il n'y a plus de contrainte dorénavant.

Finalement, aux 2427 solutions, il faut retrancher 56 cas, ce qui fait 2427 – 56 = 2371 cas. Ces cas sont à leur tour retranchés aux 3765 solutions initialement trouvées. Soit au total 3765 – 2371 = 1394 solutions au problème.

Remarque : Le problème que nous venons de traiter est lié au problème dit de la palmeraie (*livre Combien ? volume 1, chapitre 1*). Celui-ci revient à prendre N des mots précédents de façon qu'ils n'aient aucune lettre en commun, ce qui constitue une partition de l'ensemble des nombres de 0 à $N^2 - 1$. Le nombre de telles partitions, où chacune des partitions a le même poids, est de 2 pour $N = 3$ (précisément les partitions (0 4 8) (1 5 6) (2 3 7) et (0 5 7) (1 3 8) (2 4 6)), puis 392 pour $N = 4$, et 3 245 664 pour $N = 5$.