I Binôme et plus

1. Écrire une fonction `b(n:int,p:int)->int` qui renvoie la valeur du coefficient binomial $\binom{n}{p}$ en utilisant une programmation itérative (calcul de bas en haut)
2. On définit les coefficients trinômiaux (qui permettent entre autres de développer $(a+b+c)^n$) de la façon suivante : si $i+j+k=n$

$$\binom{n}{i,j,k} = \begin{cases} \binom{n}{j} & \text{si } i=0 \\ \binom{n}{i} & \text{si } j=0 \text{ ou } k=0 \\ \binom{n-1}{i-1,j,k} + \binom{n-1}{i,j-1,k} + \binom{n-1}{i,j,k-1} & \text{sinon} \end{cases}$$

   a) On considère la fonction suivante :

```
def T(i,j,k) :
    n = i+j+k
    if i == 0 :
        return b(n,j)
    elif j*k == 0 :
        return b(n,i)
    else :
        return T(i-1,j,k)+T(i,j-1,k)+T(i,j,k-1)
```

   Justifier que la complexité de cette fonction est exponentielle par rapport à l'entier $n=i+j+k$.

   b) En déduire une fonction récursive mémoïsée à partir de `T`.

   c) Écrire une fonction itérative qui renvoie aussi cette valeur. *On pourra introduire un « tableau » à trois dimensions `Pyramide` tel que `Pyramide[n][i][j]` contienne la valeur de* $\binom{n}{i,j,k}$, *où* $k=n-i-j$.

II Monter les escaliers

On considère un escalier qui possède $n \geqslant 1$ marches qui l'on souhaite monter en montant à chaque pas 1, 2, 3 ou 4 marches. On note u_n le nombre de façons de monter cet escalier.

1. Déterminer une relation entre les termes de la suite (u_n).
2. En déduire une fonction récursive « efficace » `U(n:int)->int` qui calcule la valeur de u_n.
3. À l'aide d'une fonction itérative, évaluer la complexité du calcul de u_n en fonction de n.

III Chemins dans un carré

On considère le problème suivant : on souhaite déplacer un pion depuis l'origine du repère $(0,0)$ jusqu'au point de coordonnées (N,N), où $N \in \mathbb{N}^*$ est fixé. Pour cela, on autorise les déplacements du pion horizontalement d'une case, verticalement d'une case et en diagonale (en haut et à droite) d'une case.

1. On note $f(x,y)$ le nombre de chemins reliant le point de coordonnées (x,y) et le point d'arrivée (N,N). Déterminer la valeur de $f(x,y)$ en fonction de $f(x+1,y)$, $f(x,y+1)$ et $f(x+1,y+1)$, en précisant les éventuels cas particuliers.
2. En déduire une fonction `chemins(N:int)->int` qui renvoie le nombre de chemins allant de $(0,0)$ à (N,N).