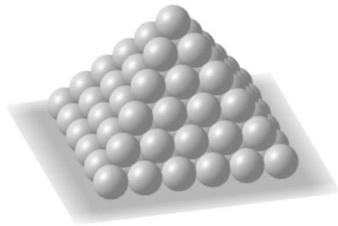
On considère une pyramide à base carrée formée de boules identiques empilées les unes sur les autres:

- le 1er étage, situé au niveau le plus haut, est composé de 1 boule ;
- le 2e étage, niveau juste en dessous, est composé de 4 boules ;
- le 3e étage possède 9 boules ;
- ...
- le n-ième étage possède n^2 boules.



Pour tout entier $n \geqslant 1$, on note u_n le nombre de boules qui composent le n-ième étage en partant du haut de la pyramide. Ainsi, $u_n = n^2$.

1. Calculer le nombre total de boules d'une pyramide de 4 étages.
2. On considère la suite (S_n) définie pour tout entier $n \geqslant 1$ par

$$S_n = u_1 + u_2 + \ldots + u_n.$$

   (a) Calculer S_5 et interpréter ce résultat.

   (b) On considère la fonction `pyramide` ci-dessous écrite de manière incomplète en langage Python. Recopier et compléter sur la copie le cadre ci-dessous de sorte que, pour tout entier naturel non nul n, l'instruction `pyramide(n)` renvoie le nombre de boules composant une pyramide de n étages.

```
def pyramide(n) :
    S = 0
    for i in range(1, n+1) :
        S = ...
    return ...
```

   (c) Vérifier que pour tout entier naturel n :

$$\frac{n(n+1)(2n+1)}{6} + (n+1)^2 = \frac{(n+1)(n+2)[2(n+1)+1]}{6}$$

   (d) Démontrer par récurrence que pour tout entier $n \geqslant 1$:

$$S_n = \frac{n(n+1)(2n+1)}{6}.$$

3. Un marchand souhaite disposer des oranges en pyramide à base carrée. Il possède 200 oranges. Combien d'oranges utilise-t-il pour construire la plus grande pyramide possible ?