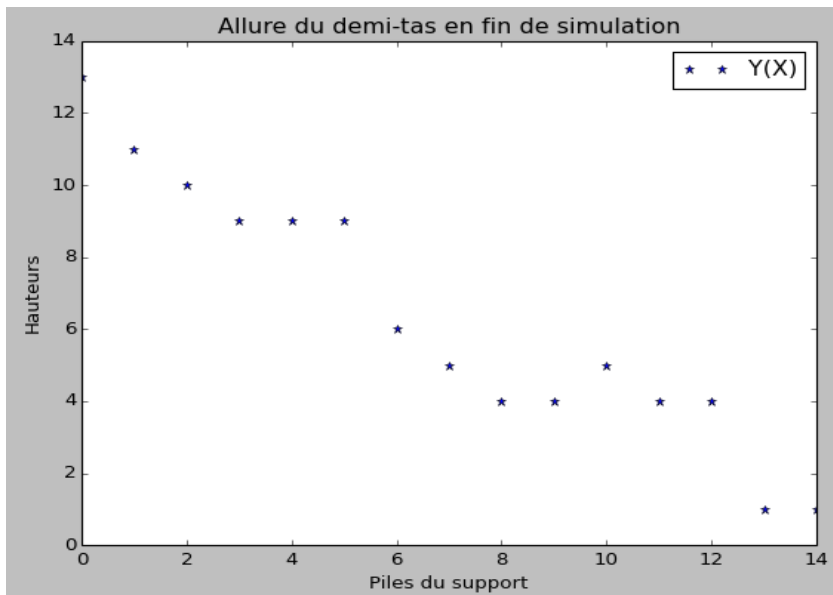


Q10



Q11

Le code définit une **classe grain**.

L'instanciation d'un objet de cette classe crée un objet avec les 3 attributs suivants :

- 1- coordonnées de position (x,y) où x et y sont des valeurs passées en arguments.
- 2- vitesse nulle
- 3- force nulles

Ceci grâce à la méthode constructeur **def __init__(self, x, y)**

D'autres méthodes de cette classe pourront ensuite agir sur les objets « grain » instanciés.

Q12. Forme du tas : « pointu » avec un pas de $2R$ en x et de $\text{racine}(3)$ en y



Points de l'exemple ci-dessus :

- * $i=0$ donne $(0,R)$ $(2R, R)$ $(4R, R)$... $((2n-1)R, R)$
- # $i=1$ donne $(R, R(1+\text{rac}(3)))$ $(3R, R(1+\text{rac}(3)))$ $(5R, R(1+\text{rac}(3)))$... $((2n-3)R, R(1+\text{rac}(3)))$
- / $i=2$ donne $(2R, R(1+\text{rac}(3)))$ $(4R, R(1+\text{rac}(3)))$ $(6R, R(1+\text{rac}(3)))$... $((2n-5)R, R(1+\text{rac}(3)))$
- ...
- $i=n-1$ donne : $(n-1)R, R(1+(n-1)*\text{rac}(3))$ → **sommet du tas \$ (j=0 uniquement)**

Q13

On peut donc écrire : $F : (x_1, x_2) \rightarrow (x_2, -\mu g - K_T/m \cdot x_1)$

$$G : (x_1, x_2) \rightarrow (x_2, -g - \gamma_N/m \cdot x_2 - K_N/m \cdot x_1)$$

Trace euler

