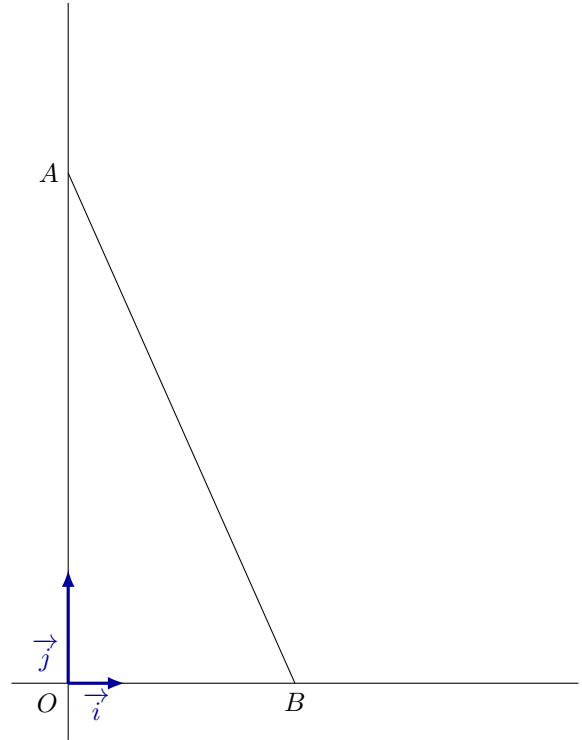
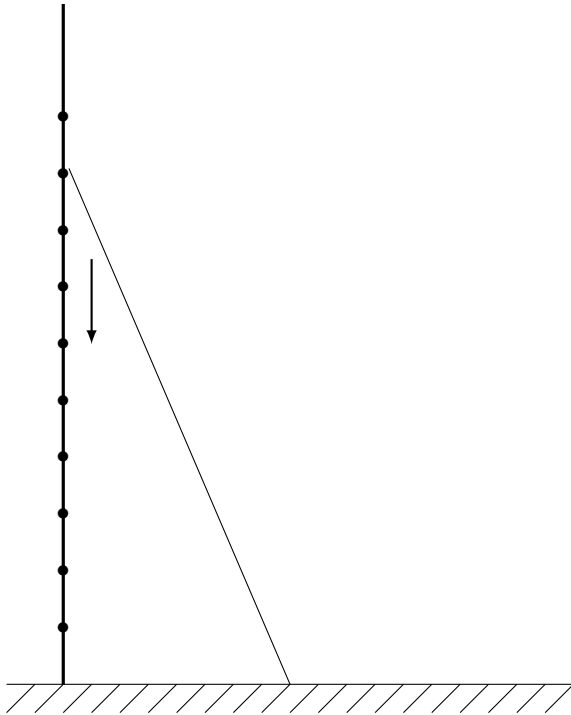


Glissement d'une échelle le long d'un mur

Une poutre de 10 mètres de haut est appuyée contre une échelle verticale. Elle est initialement verticale, puis glisse sur le sol et tombe d'échelon en échelon jusqu'à se retrouver complètement à l'horizontal sur le sol.

On note A et B les points aux extrémités de la poutre.



1. Donner les coordonnées de A et de B lorsque la poutre est verticale.
2. Donner les coordonnées de A et de B lorsque la poutre a glissé d'un échelon.
3. Donner les coordonnées de B lorsque l'ordonnée de A est y .
4. Ecrire un algorithme et un programme qui trace toutes les positions successives de la poutre, échelon après échelon, depuis sa position verticale jusqu'à sa position finale horizontale.
*Dans la bibliothèque Libxy, les commandes $A=(x_A, y_A)$, $B=(x_B, y_B)$ puis $\text{Line}(A, B)$ permettent de tracer le segment $[AB]$.
 On oubliera pas de précéder toute commande par :*

```

1 from Libxy import *
2 InitGraph ()
3
4 # Instructions python et graphiques
5
6 SaveGraph ()

```

5. Donner, en fonction des coordonnées de B , celles de B' symétrique de B par rapport à l'axe des ordonnées. Tracer alors aussi le segment $[AB']$.
6. Donner, en fonction des coordonnées de A , celles de A' symétrique de A par rapport à l'axe des abscisses. Tracer les segments $[A'B]$ et $[A'B']$.

Annexe : Diverses commandes du module Libxy

- `Point(x,y)` place le point de coordonnées (x, y) ;
- `Line(A,B)` trace le segment $[AB]$;
- `Vector(A,B)` trace le vecteur d'origine A et d'extrémité B ;
- `Polygon(A,B,C,...)` trace le polygone dont les sommets sont $A, B, C, \dots$;
- `Circle(A,R)` trace le cercle de centre A et de rayon R ;
- `Axes(xtick,ytick)` place les axes du graphiques, `xtick` et `ytick` étant les pas des graduations utilisées sur chaque axe. Par défaut, 10 graduations parcourent chaque axe ;
- `Grid(Dx,Dy)` trace un quadrillage de pas d'abscisse Dx et d'ordonnée Dy ;