

Le viaduc de Millau

Compléments pour le professeur (pour éventuellement guider la recherche si nécessaire) :

Analyse du problème :

- on peut demander aux élèves de faire le schéma d'un hauban en précisant les dimensions afin de dégager une démarche de résolution.

Décomposer en sous-problèmes :

- Calcul de la longueur d'un câble en fonction des distances a et b des points d'ancrage sur le pylône et sur le sol.

Faire un schéma (triangle rectangle)

Mobiliser le théorème de Pythagore

Ecrire une fonction informatique qui renvoie la longueur d'un câble en fonction de a et b

```
1 from math import sqrt
2 def longueurcable(a,b):
3     return sqrt(a**2+b**2)
```

On peut déjà tester en calculant la longueur du plus grand câble avec a=70 et b=150

```
>>> longueurcable(70,150)
165.5294535724685
```

- Calcul de la longueur de câble nécessaire pour un hauban :
 - on répète onze fois le même calcul avec des distances différentes → utilisation d'une boucle For
 - pour le plus grand câble, a=70 et b=150
 - puis à chaque étape, pour le câble suivant, a diminue de 2 et b diminue de 3
 - écrire une fonction qui renvoie la somme des longueurs des 11 câbles d'un hauban

```
5 def longueurcable_hauban():
6     a=70
7     b=150
8     longueurcable_hauban=0
9     for i in range(11):
10         longueurcable_hauban=longueurcable_hauban+longueurcable(a,b)
11         a=a-2
12         b=b-3
13     return longueurcable_hauban
```

- Calcul de la longueur de câble pour le pont

```
16 def longueurcable_pont():
17     return longueurcable_hauban()*2*7
```

```
>>> longueurcable_pont()
22752.815924941988
```

Réponse au problème :

Environ 22752,8 m soit plus de 22,7 km de câble ont été nécessaires pour la construction du pont !

Programme(s) codé(s) en Python

Première version (avec utilisation systématique de fonctions)

```
1 from math import sqrt
2 def longueurcable(a,b):
3     return sqrt(a**2+b**2)
4
5 def longueurcable_hauban():
6     a=70
7     b=150
8     longueurcable_hauban=0
9     for i in range(11):
10         longueurcable_hauban=longueurcable_hauban+longueurcable(a,b)
11         a=a-2
12         b=b-3
13     return longueurcable_hauban
14
15 def longueurcable_pont():
16     return longueurcable_hauban()*2*7
```

```
>>> longueurcable_pont()
22752.815924941988
```

Deuxième version (sans utilisation systématique de fonctions)

```
18 ##deuxième version sans utilisation systématique de fonctions
19 from math import sqrt
20 def longueurcable(a,b):
21     return sqrt(a**2+b**2)
22
23
24 a=70
25 b=150
26 longueurcable_hauban=0
27 for i in range(11):
28     longueurcable_hauban=longueurcable_hauban+longueurcable(a,b)
29     a=a-2
30     b=b-3
31
32
33 longueurcable_pont=longueurcable_hauban*2*7
..
```

```
>>> longueurcable_pont
22752.815924941988
```