

On dispose de 1 000 lampes numérotées de 1 à 1 000.

- à l'étape 0, toutes les lampes sont éteintes ;
- à l'étape 1, on allume toutes les lampes ;
- à l'étape 2, on change l'état des lampes dont le n° est divisible par 2 ;
- à l'étape 3, on change l'état des lampes dont le n° est divisible par 3 ;
- ...
- à l'étape n , on change l'état des lampes dont le n° est divisible par n .

Au bout des 1 000 étapes, quels sont les n° des lampes allumées ?
Émettre une conjecture, puis la démontrer !





Sortie : la liste des n° des lampes allumées

- 1 créer la liste L des 100 lampes
 - 2 **pour** j **to** 1 **jusque** 100 **faire**
 - 3 | parcourir la liste L :
 - 4 | **si** le n° de la lampe est divisible par j **alors**
 - 5 | | changer l'état de lampe
 - 6 Créer la liste des n° des lampes allumées
-

Comme la notion de diviseur est compliquée en 2^{nde} on peut compter de j en j .

Sortie : la liste des n° des lampes allumées

- 1 créer la liste L des 100 lampes
 - 2 **pour** j **to** 1 **jusque** 100 **faire**
 - 3 | parcourir la liste L de j en j :
 - 4 | changer l'état de lampe
 - 5 Créer la liste des n° des lampes allumées
-

Il faut ensuite modéliser l'état de la lampe, 1 pour allumé et 0 pour éteint par exemple.

Sortie : la liste des n° des lampes allumées

- 1 créer la liste L des 100 lampes
 - 2 **pour** j **to** 1 **jusque** 100 **faire**
 - 3 | parcourir la liste L de j en j :
 - 4 | **si** le n° de la lampe est 0 **alors**
 - 5 | | il devient 1
 - 6 | **sinon**
 - 7 | | il devient 0
 - 8 Créer la liste des n° des lampes allumées
-

```
1  # -*- coding:utf8 -*-
2  # python 3
3
4  """ On dispose de 1000 lampes numérotées de 1 à 1000.
5  - à l'étape 0, toutes les lampes sont éteintes ;
6  - à l'étape 1, on allume toutes les lampes ;
7  - à l'étape 2, on change l'état des lampes dont le n° est
      ↪ divisible par 2 ;
8  - à l'étape 3, on change l'état des lampes dont le n° est
      ↪ divisible par 3;
9  ...
10 - à l'étape n, on change l'état des lampes dont le n° est
      ↪ divisible par n.
11 Au bout des 1000 étapes, quels sont les n° des lampes allumés
      ↪ es ?
12 Emettre une conjecture, puis la démontrer !
13 """
14
15 # 0 = lampe éteinte / 1 = lampe allumée
16 # à la première étape toutes les lampes sont allumées
17 lampes = [1 for k in range(1000)]
18 for n in range(2,1001): # range = [2;1001[
19     k = n - 1 #les indices commencent à 0
20     while k < 1000:
21         lampes[k] = 1 - lampes[k] # changement d'état
22         k = k + n
23 allume = [i + 1 for i, etat in enumerate(lampes) if etat == 1
      ↪ ]
24 print("les n° des lampes allumées sont :",allume)
```



Conjecture : les lampes allumées sont celles dont le n° est un carré.

Démonstration : soit n le numéro de la lampe.

- Le nombre de changement d'état est le nombre de diviseurs de n .
- Si la lampe est allumée, alors la lampe a changé d'état un nombre impair de fois.

Seuls les carrés parfaits ont un nombre impairs de diviseurs.

4 a pour diviseurs	4	2
	1	

36 a pour diviseurs	36	18	12	9	6
	1	2	3	4	

15 a pour diviseurs	15	5
	1	3