

```

1  #version itérative naïve
2  def vol(p):
3      u=p
4      compteur=0
5      while u!=1:
6          if u%2==0:
7              u=u//2
8          else:
9              u=3*u+1
10             compteur+=1
11
12     return compteur
13
14 #version récursive naïve
15 def vol_rec(p):
16     if p==1:
17         return 0
18     if p%2==0:
19         return vol_rec(p//2)+1
20     else:
21         return vol_rec(3*p+1)+1
22
23
24 maxi=vol(1)
25 for p in range(2,10**3+1):
26     valeur=vol(p)
27     if valeur>maxi:
28         maxi=valeur
29 print('le temps de vol maximum pour p entre 1 et 1000 est: ',maxi)
30
31 #On obtient 178
32
33 #Pour prendre des valeurs de p plus grandes, on va utiliser la mémorisation:
34
35 D={1:0} #dictionnaire où on va stocker les temps de vol
36     #pour p=1, le temps de vol est de 0
37
38 def vol_mem(p):
39     if p in D:
40         return D[p]
41
42     if p%2==0:
43         D[p]=vol_mem(p//2)+1
44         return D[p]
45     else:
46         D[p]=vol_mem(3*p+1)+1
47         return D[p]
48
49
50
51 maxi=vol_mem(1)
52 for p in range(2,10**7+1):
53     valeur=vol_mem(p)
54     if valeur>maxi:
55         maxi=valeur
56 print('Avec memoisation, le temps de vol maximum pour p entre 1 et 10**7 est: ',maxi)
57
58 #On obtient 685
59
60 compteur=0
61 for p in range(1,10**6+1):
62     if vol_mem(p)<=30:
63         compteur+=1
64 print('Le nombre d''entiers p entre 1 et 10**6 vérifiant vol(p)<=30 est: ',compteur)
65
66 #On obtient 2882
67
68

```