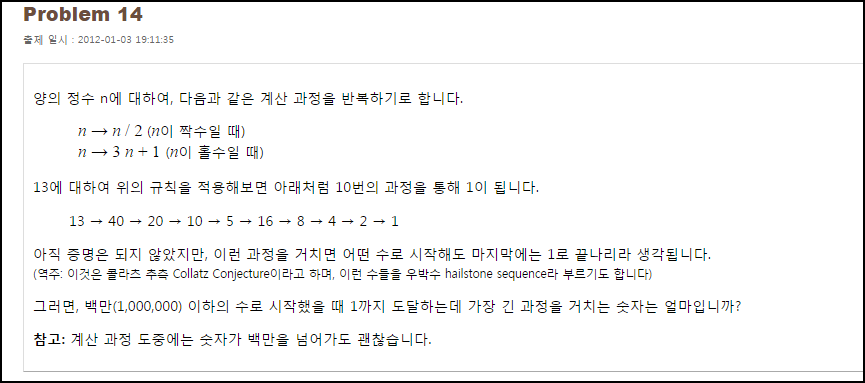
**Exercise**

우박수에 대해서 정의하고, 100만 이하의 수 중, 계산 과정이 가장 긴 우박수를 찾는 문제입니다.



**Solution\_code**

주어진 공식들 그대로 사용하고, 이전 과정과 현재 카운트를 비교해서, 클 경우 수와 카운트를 담아 둡니다.

|  |  |  |
| --- | --- | --- |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27** | **import time**  **start\_time = time.time()**  **cnt = 1**  **k= 1**  **temp = 0**  **last\_cnt = 0**    **for k in range(1,1000001):**  **cnt = 1**  **n = k**  **while (n != 1):**  **#print "[+] ",n**  **cnt += 1**  **if n%2 ==0 :**  **n /= 2**  **else:**  **n = 3\*n+1**    **if last\_cnt < cnt :**  **last\_cnt = cnt**  **temp = k**  **# k += 1**    **print "[+]" ,k, last\_cnt, temp**  **last\_time = time.time()**  **print "[+]",last\_time-start\_time** | [cs](http://colorscripter.com/info#e) |

**Result**

공식없이, 일일이 100만 까지 해당 경우의 수들을 카운트 해야해서 속도가 조금 안나오는 것 같지만 525번의 과정을 거쳐야 1이 나오는 우박수인 837799를 구할 수 있었습니다.

