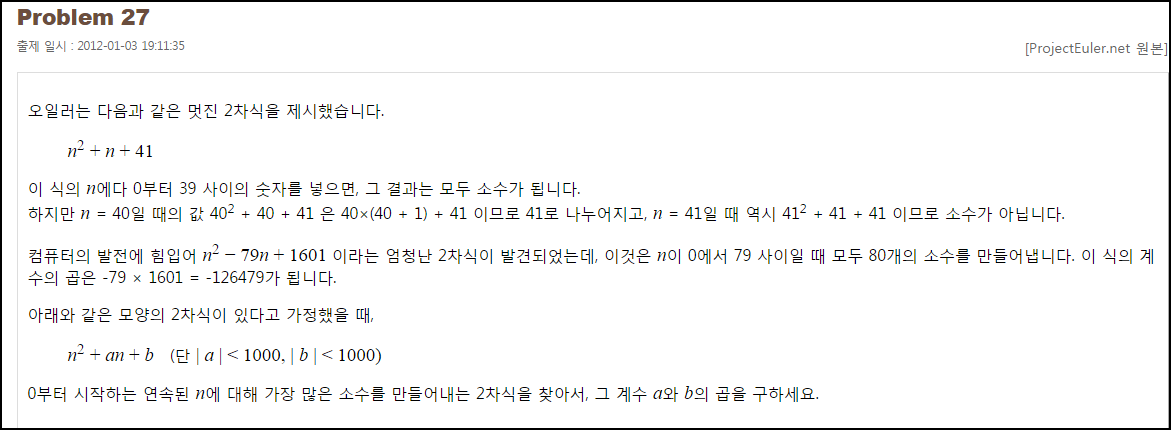
**Exercise**

연속되는 n 값에 따라 가장 많은 소수가 발생되는 2차식의 경우를 구하는 문제입니다.



**Solution\_code**

소수 판별에는 에라스토테네스의 체를 이용했고, b 의 조건은 무조건 소수여야 하는 것을 이용해 2차식을 그대로 판별했습니다.

|  |  |  |
| --- | --- | --- |
| **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**  **28**  **29**  **30**  **31**  **32**  **33**  **34**  **35**  **36**  **37**  **38**  **39**  **40**  **41**  **42**  **43**  **44**  **45**  **46**  **47** | **import time**    **start\_time = time.time()**  **def Era(N):**  **sosu = set()**  **sieve = {}**  **for i in range(2, N+1):**  **sieve[i] = 0**    **# Sieve of Eratosthenes**  **for i in range(2, N+1):**  **if sieve[i]==0:**  **n=2**  **while i\*n <= N:**  **sieve[i\*n]=1**  **n+=1**  **for i in range(2, N+1):**  **if sieve[i]==0:**  **sosu.add(i)**    **return list(sosu)**    **era = Era(10000)**  **n = 0**  **y = 0**  **cnt = 0**  **result = 0**  **b\_list = Era(1000)**  **result\_a = 0**  **result\_b = 0**    **for a in range(-1000,1000):**  **for b in b\_list:**  **while(True):**  **y = n\*\*2+a\*n+b**  **if y in era:**  **n+=1**  **else:**  **if result < n:**  **result = n**  **result\_a = a**  **result\_b = b**  **n=0**  **break**    **print "[+]",result,result\_a,result\_b, result\_b\*result\_a**  **print "[+]",time.time()-start\_time** | [cs](http://colorscripter.com/info#e) |

**Result**

