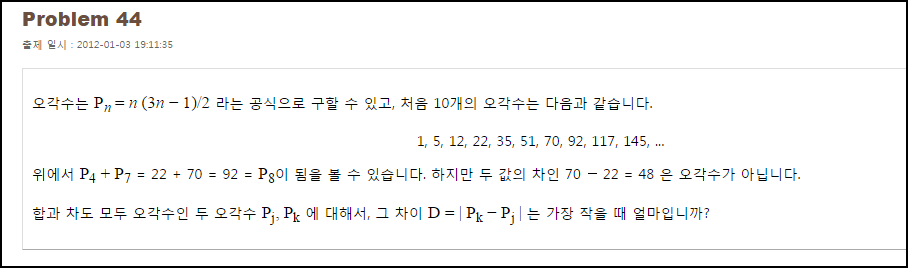
**Exercise**

오각수에 대해서 정의하고, 오각수 중에서 두 수의 합도 오각수, 차도 오각수인 경우에 대해 절대 값 Pk-Pj 의 값을 구하는 문제입니다.



**Solution\_code**

먼저, 적절한 범위의 오각수 리스트를 만들고, 무식하게, for 문 돌면서 안에서 오각수 리스트 내에 원소가 있는지 여부를 이용해서 판별 했었으나… 비효율적으로 시간이 오래 걸리길래, 오각수 판별 함수를 이용해서 시간을 단축 시킬 수 있었습니다. 상당

|  |  |  |
| --- | --- | --- |
| **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** | **========1번째**  **F\_num = []**    **for i in range(1,10000):**  **F\_num.append(i\*(3\*i-1)/2)**    **for i in F\_num:**  **for j in F\_num:**  **if i+j in F\_num and (i-j in F\_num or j-i in F\_num):**  **print "[+]",i,j,abs(i-j)**      **========2번째**    **import math**    **def Check(n):**  **n = abs(n)**  **p = ( math.sqrt(1 + 24\*n) + 1 ) / 6**  **return p==int(p)**    **F\_num = []**  **for i in range(1,10000):**  **F\_num.append(i\*(3\*i-1)/2)**    **for i in F\_num:**  **for j in F\_num:**  **if Check(i+j) and (Check(i-j) or Check(j-i)):**  **print "[+]",i,j,abs(i-j)**      ***[Colored by Color Scripter](http://colorscripter.com/info#e)*** | [cs](http://colorscripter.com/info#e) |

**Result**

