Fibonacci All Pairs (COJ 2828)

Frank Arteaga Salgado
farteaga@ipvce.lt.rimed.cu

May 28, 2016

1 Problema

Vea este problema en el Juez Caribeño en Línea, 2828 - Fibonacci All Pairs. Pero antes de enfrentarse a este problema conviene resolver primero los problemas 1596 - Fibonacci Numbers y 1872 - Cousin Party.

1.1 Enunciado

La Secuencia de Fibonacci se define recursivamente como:

$$\begin{aligned} F_1 &= 1 \\ F_2 &= 2 \\ F_n &= F_{n-1} + F_{n-2} \qquad n > 2 \end{aligned} \tag{1}$$

Considere todos los pares ordenados de valores distintos de los k primeros términos de la Secuencia de Fibonacci, esto es, (F_i, F_j) con $1 \leq i < j \leq k$. Se desea calcular la suma de los productos de cada uno de estos pares:

$$S(k) = \sum_{1 \leq i < j \leq k} F_i \times F_j$$

Ejemplo, para $k = 4$ tenemos $F_1 = 1, F_2 = 2, F_3 = 3$ y $F_4 = 5$; los productos de los pares ordenados son $F_1 \times F_2 = 2, F_1 \times F_3 = 3, F_1 \times F_4 = 5, F_2 \times F_3 = 6, F_2 \times F_4 = 10, F_3 \times F_4 = 15$ y su suma es $2 + 3 + 5 + 6 + 10 + 15 = 41$.

2 Solución

Si implementamos dos ciclos anidados como sugiere el enunciado del problema entonces obtendríamos una solución $\Theta(k^2)$, la cual es muy costosa por los rangos de k[1].

Notemos que si sumamos cada par dos veces y añadiéramos $\sum_{i=1}^{k} F_i^2$ entonces nos quedaría:

$$2S(k) + \sum_{i=1}^{k} F_i^2 = \sum_{1 \leq i,j \leq k} F_i \times F_j = \left(\sum_{i=1}^{k} F_i \right)^2 \tag{2}$$

[1] $k \leq 10^9$

De donde se despeja $S(k)$:

$$S(k) = \frac{\left(\sum_{i=1}^{k} F_i\right)^2 - \sum_{i=1}^{k} F_i^2}{2} \tag{3}$$

Hasta aquí la expresión (3) nos permite una implementación con costo lineal $\Theta(k)$, pero todavía no es suficiente para los rangos del problema. Por lo que es necesario encontrar expresiones para las dos sumatorias del numerador que tengan algoritmos más eficientes que el lineal para computarlas.

Esas expresiones son:

$$\sum_{i=1}^{k} F_i = F_{k+2} - 2 \tag{4}$$

$$\sum_{i=1}^{k} F_i^2 = F_k \times F_{k+1} - 1 \tag{5}$$

Las cuales el lector puede demostrar fácilmente por inducción. Luego de esto sustituimos (4) y (5) en (3):

$$S(k) = \frac{(F_{k+2} - 2)^2 - F_k \times F_{k+1} + 1}{2} \tag{6}$$

Lo que redujo el cómputo de $S(k)$ al cálculo de tres números de Fibonacci. Para calcular F_n eficientemente en tiempo $O(log\ n)$ se usa multiplicación de matrices explotando el siguiente hecho[2]

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n = \begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} \tag{7}$$

cuando $F_0 = 0$ y $F_1 = 1$, que son condiciones iniciales distintas a la definición (1), sin embargo emerge la misma secuencia de números, pero desplazada como se muestra en la tabla:

F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7
1	1	2	3	5	8	13	21
0	1	1	2	3	5	8	13

[2]Igual que las identidades (4) y (5) le recomendamos al lector demuestre (7) por inducción.