fibonacci

⋮

```
while (fibM < n) {    - - - -    (1)
        cont += 3
}
    ⋮
while (fibM > 1) {

    if (arr[i] < buscar) {
            cont += 4
    }
    else if (arr[i] > buscar) {
            cont += 3
    }
    else
        return i;
}
            cont++
if (fibM1 && arr[inicio+1] == busca
```

Cada que entra en el while (1) es la posición del termino de la serie de Fibonacci

La formula de Binet y Moivre nos da el termino m

$$F_m = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^m - \left(\frac{1-\sqrt{5}}{2}\right)^m}{\sqrt{5}}, \qquad m = \text{posición}$$

Pero nos interesa saber la posición, para ello despejamos m.
Como es dificil despejarlo de esta forma consideramos lo siguiente

Para m muy grandes, se comporta cada termino como

$$\lim_{m \to \infty} \left(\frac{1+\sqrt{5}}{2}\right)^m = \infty$$

$$\lim_{m \to \infty} \left(\frac{1-\sqrt{5}}{2}\right)^m = 0$$

$\therefore$ el segundo termino es despreciable

$\therefore$ nos queda de la siguiente forma

$$F_m \approx \frac{\left(\frac{1+\sqrt{5}}{2}\right)^m}{\sqrt{5}}$$

Despejando

$$\sqrt{5}\, F_m = \left(\frac{1+\sqrt{5}}{2}\right)^m$$

$$\ln(\sqrt{5}\, F_m) = \ln\left(\frac{1+\sqrt{5}}{2}\right)^m$$

$$\ln(\sqrt{5}) + \ln(F_m) = m \ln\left(\frac{1+\sqrt{5}}{2}\right)$$

$$\boxed{m = \frac{\ln(\sqrt{5}) + \ln(F_m)}{\ln\left(\frac{1+\sqrt{5}}{2}\right)}}$$

$$A = \ln(\sqrt{5})$$

sea

$$B = \frac{1}{\ln\left(\frac{1+\sqrt{5}}{2}\right)}$$

- Mejor caso: elemento a buscar, es el primer elemento que comparamos

$$F_t = (A + B\ln(n) - \underset{\uparrow}{1}) + 3$$

ya pasó por la 1ra posición

Si solo se toma las instrucciones de cada if

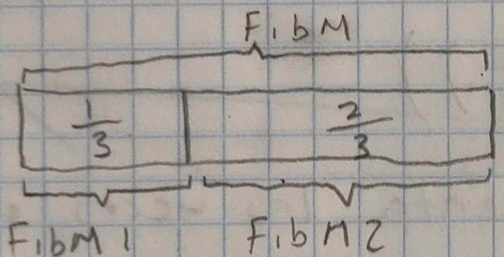
$$f_t = \underline{O(1)}$$

- Peor caso

Que no exista y sea mayor al número más grande del arreglo

e.g $FibM = FibM_1 + FibM_2$

$5 = 2 + 3$



aquí se encontraría el número

se encuentra en la $\frac{2}{3}$ parte

$$F_t = \underbrace{(A + B\ln(n) - 1)}_{\text{while}} + 4\underbrace{\left(A + B\ln\left(2\frac{n}{3}\right)\right)}_{\text{1er if}}$$

$$F_t = \underline{5A + B\left(\ln(n) + \ln\left(\frac{2n}{3}\right)\right) - 1}$$

Si solo se toma las instrucciones ó cuantas veces entra en cada if

$$f_t = O(\log n)$$

- Caso medio

tomar ahora en cuenta el otro if

(1) $(A + B\ln(n) - 1) + 3\left(\left(A + B\ln(2n)\right) - \left(A + B\ln\left(\frac{2n}{3}\right)\right)\right)$

while

decrementa 2 pasos mas

Le quitamos la $\frac{2}{3}$ partes

Sumando cada caso y tomando que cada caso es equiprobable

$$f_t = \frac{1}{3}(\text{Mejor caso} + \text{Peor caso} + (1))$$

$$f_t = \frac{1}{3}\left(7A + 2B\ln(n) - B\left(\ln\left(\frac{2n}{3}\right) + \ln(n)\right) + 3\left(B\ln(2n) - B\ln\left(\frac{2n}{3}\right)\right)\right)$$

si solo se toma en cuenta las veces que entra en cada if

$$f_t = O(\log n)$$