

## Serie 1 - Serie Fibonacci

- Joaquin Sasnalema

La serie inicia con dos valores base:

$$\begin{aligned} a &= 0 \\ b &= 1 \end{aligned} \rightarrow c = a + b$$

Después de obtener el valor, actualizo las variables:  
a toma el valor de b.  
b toma el valor de a

Se repite este proceso tantas veces como términos quiera generar.

## Serie 2 - Impares intercalados con ceros.

La serie sigue un patrón fijo que se repite siempre:

- 1- Un número impar
- 2- Un cero
- 3- Otro número impar

- 4- Otro cero
- 5- ... y así sucesivamente

Los números impares aumentan de dos en dos: 1, 3, 5, 7, 9...

El cero siempre aparece entre cada par de impares.

```
if (1 % 2 == 0) {  
    System.out.println (imp);  
    imp += 2;  
  
else {  
    System.out.println (0);  
}
```

## Serie 3 - Serie de fracciones formadas por Fibonacci en el numerador e impares consecutivos en el denominador.

La serie tiene dos secuencias trabajando al mismo tiempo:

- 1) El numerador es Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13...
- 2) El denominador son números impares consecutivos: 1, 3, 5, 7, 9, 11, 13, 15...

```
System.out.println (a + "/" + imp + " ");  
imp += 2;  
int c = a + b; a = b; b = c;
```

## Serie 1 - Serie Fibonacci

- Joaquin Sosna Lema

La serie inicia con dos valores base:

$$a=0$$
$$b=1$$

$$c = a + b$$

Después de obtener el valor, actualizo las variables:  
a toma el valor de b.  
b toma el valor de c

Se repite este proceso tantas veces como términos quiera generar.

## Serie 2 - Impares intercalados con ceros.

La serie sigue un patrón fijo que se repite siempre:

- 1 - Un número impar
- 2 - Un cero
- 3 - Otro número impar

- 4 - Otro cero
- 5 - ... y así sucesivamente

Los números impares aumentan de dos en dos: 1, 3, 5, 7, 9...

El cero siempre aparece entre cada par de impares.

```
if (1 % 2 == 0) {  
    System.out.println (imp);  
    imp += 2;  
}
```

```
else {  
    System.out.println (0);  
}
```

## Serie 3 - Serie de fracciones formadas por Fibonacci en el numerador e impares consecutivos en el denominador.

La serie tiene dos secuencias trabajando al mismo tiempo:

1) El numerador es Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13...

2) El denominador son números impares consecutivos: 1, 3, 5, 7, 9, 11, 13, 15...

```
System.out.println (a + "/" + imp + " ");
```

impar += 2;

```
int c = a + b; a = b; b = c;
```

Serie 4 - Serie de fracciones con Fibonacci en el numerador y números pares consecutivos en el denominador.

1) Numerador → Fibonacci

0, 1, 1, 2, 3, 5, 8, 13...

2) Denominador → Pares consecutivos

2, 4, 6, 8, 10, 12, 14, 16...

```
System.out.println(a + "/" + par + " ");  
par += 2;  
int c = a + b;  
a = b;  
b = c;
```

Serie 5 - Números primos consecutivos.

La serie muestra solo números primos, uno tras otro: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31...

Un número primo es aquel que solo se puede dividir entre 1 y él mismo sin dejar residuo.

```
boolean primo = true;  
for (int j = 2; j < suma; j++) {  
    if (num % j == 0) {  
        primo = false;  
        break;  
    }  
}
```

Nombre de la serie 56 - Cuadrados perfectos.

La serie general el cuadrado de cada número número natural:

$1^2 = 1$  Así va creciendo; cada término, el número multiplicado

$2^2 = 4$  por sí mismo.

$3^2 = 9$  int cuadrado = i \* i;

```
System.out.println(cuadrado + " ");
```

Serie 7 - Aritmética de diferencia constante 3.

La serie crece siempre sumando 3 al término anterior.

1

$1 + 3 = 4$

```
System.out.println(valor + " ");
```

$4 + 3 = 7$

Valor += 3;

$7 + 3 = 10$

$10 + 3 = 13$

$13 + 3 = 16$

Jhoan Sasnalema

## Serie 42 - No aritmética

Jhoan Sas nálema

No es aritmética porque la diferencia no es constante, y no es geométrica porque no se multiplica por un número fijo.

Diferencia entre términos:

$$6-2 = 4$$

$$12-6 = 6$$

$$20-12 = 8$$

$$30-20 = 10$$

$$42-30 = 12$$

Empieza en 2, cada siguiente término se obtiene sumando números pares consecutivos

$$a_{n+1} = a_n + (2n + 2).$$

int valor = 2;

int incremento = 4;

```
for (int i=1; i <= n; i++) {  
    System.out.print(valor + " ");  
    valor += incremento;  
    incremento += 2; } {
```