
TALLER GRUPAL N°1

Integrantes: Jhoel Ordoñez

Jandry Jaramillo

Julio Cesar Luna

Kriss Bustamante

Adrian Rivera

Luis Granda

Tiempo transcurrido:

Repositorio: https://github.com/Jcluna05/TallerGrupal1_Grup3.git



Usuarios de GitHub:

- aarivera7
- JAOC1930
- KrissBustamante
- JandryJaramillo
- JoLunaOS
- luisgranda16

Código fuente:

```
import java.util.concurrent.ThreadLocalRandom
import java.util.stream.IntStream

println("-----")
println("Listado de Numeros")
val numbers = IntStream.generate(() => ThreadLocalRandom.current().nextInt(origin = 10_000, bound = 1_000_000)).limit(maxSize = 25).boxed().toList
numbers.stream().forEach(println)
```



Listado de los números generados

```
-----  
Listado de Numeros  
837805  
119294  
428144  
996726  
733844  
24879  
303558  
295817  
776132  
53728  
447893  
322689  
865318  
646822  
860751  
378059  
724868  
663146  
811081  
280550  
180024  
46743  
487211  
977319  
223382
```

Listado de los números generados ascendente

```
println("-----")  
println("Numeros descendentes")  
numbers.stream().sorted((x, y) => y.compareTo(x)).forEach(println)
```

```
-----  
Numeros ascendentes  
24879  
46743  
53728  
119294  
180024  
223382  
280550  
295817  
303558  
322689  
378059  
428144  
447893  
487211  
646822  
663146  
724868  
733844  
776132  
811081  
837805  
860751  
865318  
977319  
996726
```



Listado de los números generados descendentemente

```
println("-----")
println("Numeros ascendentes")
numbers.stream().sorted().forEach(println)
```

```
-----
Numeros descendentes
996726
977319
865318
860751
837805
811081
776132
733844
724868
663146
646822
487211
447893
428144
378059
322689
303558
295817
280550
223382
180024
119294
53728
46743
24879
```

Listado de los números generados que son pares

```
println("-----")
println("Numeros Pares")
numbers.stream().filter(x => x % 2 == 0).forEach(println)
```

```
-----
Numeros Pares
119294
428144
996726
733844
303558
776132
53728
865318
646822
724868
663146
280550
180024
223382
```

Listado de los números generados que son impares

```
println("-----")
println("Numeros Impares")
numbers.stream().filter(x => x % 2 != 0).forEach(println)
```

Numeros Impares

837805

24879

295817

447893

322689

860751

378059

811081

46743

487211

977319

Listado de los números generados que son perfectos

```
println("-----")
println("Numeros Perfectos")
numbers.stream().filter(x => (1 until x).filter(y => x % y == 0).sum == x).forEach(println)
```

Numeros Perfectos



Listado de los números generados que son abundantes

```
println("-----")
println("Numeros Abundantes")
numbers.stream().filter(x => (1 until x).filter(y => x % y == 0).sum > x).foreach(println)
```

```
-----
Numeros Abundantes
996726
303558
776132
53728
180024
```

Listado de los números generados que son primos

```
println("-----")
println("Numeros Primos")
numbers.stream().filter(x => !(2 until x).exists(y => x % y == 0)).foreach(println)
```

```
-----
Numeros Primos
447893
811081
487211
```



Listado de los números generados que son deficiente

```
println("-----")
println("Numeros Deficienntes")
numbers.stream().filter(x => (1 until x).filter(y => x % y == 0).sum < x).forEach(println)
```

```
-----
Numeros Deficienntes
837805
119294
428144
733844
24879
295817
447893
322689
865318
646822
860751
378059
724868
663146
811081
280550
46743
487211
977319
223382
```

¿Desarrollaron alguna actividad de forma concurrente? ¿Cómo? ¿En qué actividad o actividades?

Si, porque en el desarrollo de este trabajo grupal todos buscamos una manera de resolver los problemas planteados y al mismo tiempo planteamos diferentes soluciones en Scala, luego debatimos cual fue la mejor solución y ese fue el que se puso en la presentación.



En alguna actividad o actividades
¿usaron programación paralela?
Explique su
respuesta

Si, porque en ciertos momentos
cada uno se enfocó en un rol en
especifico en el desarrollo del
código, algunos resolvieron los
problemas propuestos, otros
investigaron el fundamento
teórico de los problemas y otro
las diapositivas.

¿Algún miembro del grupo
organizó cómo resolver el taller?
¿Qué hizo? ¿Fue
efectivo?

Dos integrantes asumieron el
liderato del grupo, Adrián Rivera
se enfocó en dirigir los problemas
del código, y Julio Luna se dedico
a resolver inconvenientes con el
código y organizar la entrega del
trabajo.

