



## (6015) - Paradigmas de Programación - C1 - 2024/2

[Aula](#) [Participantes](#) [Calificaciones](#)[General](#)[Clase 3](#)[Clase 1](#)[Clase 2](#)

### Hola

Esta semana nos toca parar la pelota un poco y hacer un repaso de todo lo que aprendimos en las clases previas. Obviamente también vamos a aprender cosas nuevas del lenguaje de JavaScript pero el foco principal será ir preparándonos para el primer parcial que será el 3 de Octubre.

En esta sección vas a encontrar el enunciado de ejercicio que trabajamos en clase y todo el material complementario que es muy, mucho importante que lo leas y repases en tus momentos de estudio durante la semana,

[Ejercicio - Producción de una Máquina](#)

### Registro de producción de una máquina

Se nos pide modelar el registro de producción de una máquina. Debe registrarse la cantidad de piezas producidas por la máquina en cada día desde que se puso en operación.

Entra y lee el enunciado completo en esta sección.

[Solución del Ejercicio](#)

El archivo javascript que les entrego tiene la solución de cada uno de los puntos del enunciado del ejercicio.

Recuerden que si quieren ejecutarlo en sus computadoras tiene que tener instalado nodeJS. lo pueden descargar de aquí: <https://nodejs.org/en/download/package-manager>

Para windows la forma mas rápida de instalar usando la opción. [Prebuilt Installer](#)

### Download Node.js®

Download Node.js the way you want.

[Package Manager](#) [Prebuilt Installer](#) [Prebuilt Binaries](#) [Source Code](#)

Una vez instalado, tiene que abrir una terminal de powerShell o cmd y ejecutar el comando

node solucion.js

```
gera@MacBook-Pro-de-Infra:~/dev
~/dev-pers/test node solucion.js
algunDiaSeProducto
true
false
bitCount = 3
maximoValorDeProduccion
62
profile = async (f) => {
maximoValorDeProduccionAragana()
62
f()
```

```
valoresDeProduccionParesnow()  
[ 18, 62 ]
```

```
produccionEsAcotada  
true  
false
```

```
produccionesSuperioresA  
[ 43, 49, 62, 39 ]
```

```
produccionesSumando  
[ 48, 23, 54, 67, 38, 44 ]
```

```
produccionesSumandoObjeto
```



#### El operador spread

El operador **spread** (o de **propagación**) te permite expandir un objeto iterable en una serie de objetos individuales.

Ya sé, esta definición no es del todo clara -pero es lo mejor que se me ocurrió 😊-. Tranquilo. Veremos ejemplos que te ayudarán a dominar este operador.

⚠ Antes de empezar: la sintaxis de este operador son 3 puntos ...

