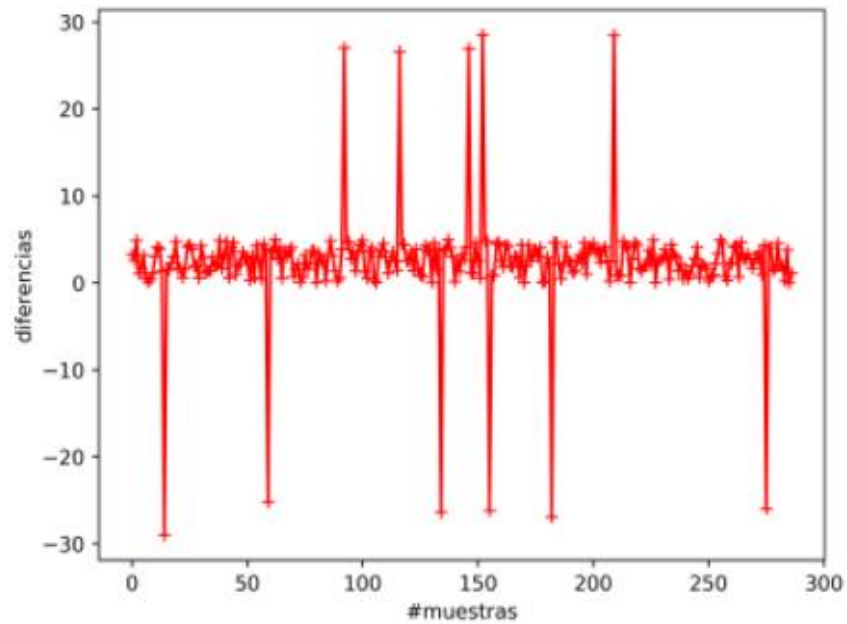


Pregunta 1:

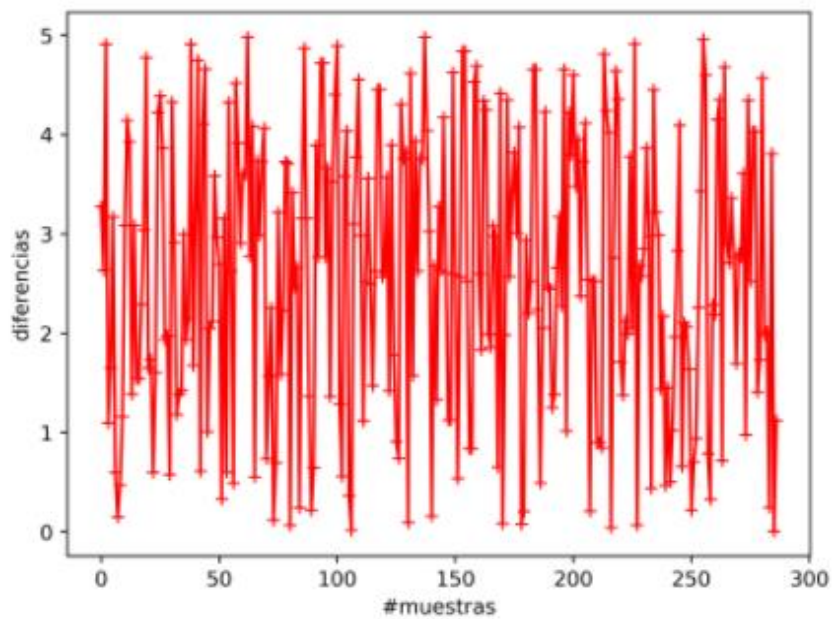
Pregunta C y D:

-Gráfico de los valores registrados del medidor para el departamento 2:



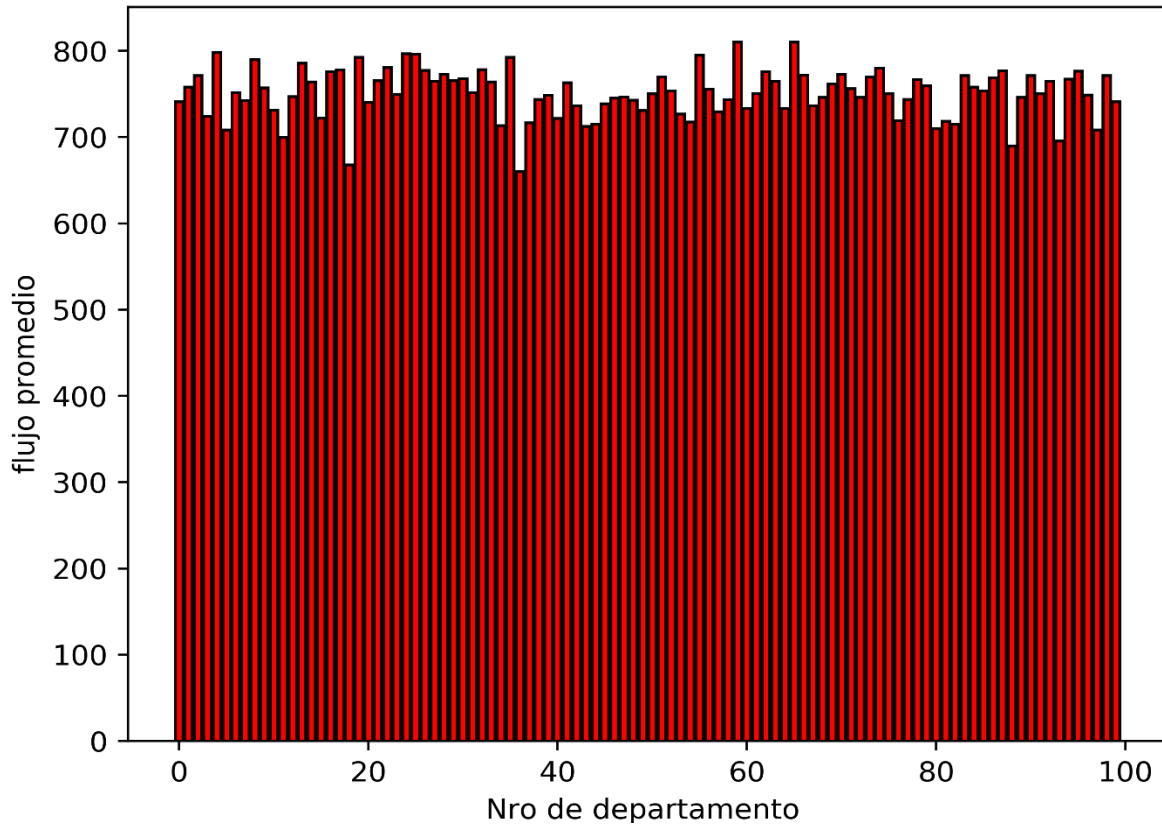
→ Cantidad de valores erróneos: 11

-Gráfico de los valores registrados del medidor corregido usando el “filtro mediana” para el departamento 2:



Pregunta E:

-Histograma del flujo promedio de cada departamento:



Pregunta 2:

Pregunta A:

```
antonio@DESKTOP-69IOMQV:~$ python3 Pregunta2.py
El tiempo para leer el archivo fue de: 0.0025046450027730316 msec
-----
```

Pregunta B:

-Cálculo de un valor de referencia para determinar el número de reinicios:

```
26     for i in range(31919):
27         valor_1 = datos[i+1+a].split(",")
28         valor_2 = datos[i+2+a].split(",")
29         a = 0
30
31         diferencia = (int(valor_2[0]) - int(valor_1[0]))
32
33         if (abs(diferencia) < 2500): #Valor referencial para determinar el numero de reinicios
34             valores.append(abs(diferencia))
35             contador = contador + 1
36             a = a + 1
37         valores.append(abs(diferencia))
38
39     print(f"Valor de referencia para determinar el numero de reinicios : {max(valores)}")
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN **TERMINAL** bash

```
antonio@DESKTOP-69IOMQV:~$ python3 Pregunta2.py
Valor de referencia para determinar el numero de reinicios : 2338
```

-Número de reinicios que se registran en cada rueda:

```
antonio@DESKTOP-69IOMQV:~$ python3 Pregunta2.py
El encoder de la rueda 1 ha presentado: 216 reinicios
El encoder de la rueda 2 ha presentado: 204 reinicios
El encoder de la rueda 3 ha presentado: 208 reinicios
El encoder de la rueda 4 ha presentado: 206 reinicios
El encoder de la rueda 5 ha presentado: 211 reinicios
El encoder de la rueda 6 ha presentado: 206 reinicios
-----
```

Pregunta C:

```
antonio@DESKTOP-69IOMQV:~$ python3 Pregunta2.py
El tiempo para escribir el archivo fue de: 0.0026209559982817154 msec
```

-Archivo creado: data_encoders.csv

```
data_encoders.csv
1  encoder1,encoder2,encoder3,encoder4,encoder5,encoder6
2  13284,13823,13101,15015,24469,24529
3  13284,13823,13101,15015,24469,24529
4  13284,13823,13101,15015,24469,24529
5  13284,13823,13101,15015,24469,24529
6  13284,13823,13101,15015,24469,24530
7  13284,13823,13101,15015,24469,24529
8  13284,13823,13101,15015,24469,24530
9  13284,13823,13101,15015,24469,24529
10 13284,13823,13101,15015,24469,24529
```

-Resultados obtenidos en terminal:

```
El tiempo para leer el archivo fue de: 0.002532771999540273 msec
-----
El encoder de la rueda 1 ha presentado: 216 reinicios
El encoder de la rueda 2 ha presentado: 204 reinicios
El encoder de la rueda 3 ha presentado: 208 reinicios
El encoder de la rueda 4 ha presentado: 206 reinicios
El encoder de la rueda 5 ha presentado: 211 reinicios
El encoder de la rueda 6 ha presentado: 206 reinicios
-----
El tiempo para escribir el archivo fue de: 0.0024170739998226054 msec
```

Pregunta D:

El tiempo de lectura mide la rapidez con la que el programa puede leer o acceder a los archivos almacenados en Visual Studio Code. Por otro lado, el tiempo de escritura mide la rapidez con la que el programa puede escribir un archivo almacenado en Visual Studio Code. Asimismo, se utiliza con más frecuencia cuando intenta copiar un archivo de una ubicación a otra y si estos archivos son enormes, cuanto más rápida sea la velocidad de lectura, menor será el tiempo de escritura. Sin embargo, esto no se cumple para los tiempos de lectura y escritura obtenidos en el archivo "data_encoders.txt", debido a que el programa no lo considera como un archivo enorme. Por esta razón, no se logra apreciar una diferencia notable entre los tiempos de escritura y lectura del archivo "data_encoders.txt".