

# Introducción a la computación

## Taller Flash N° 3

**Fecha de la primera entrega:** Viernes 13 de Mayo de 2016 9:30 hs.

**Fecha de la segunda entrega:** Miércoles 18 de Mayo de 2016 9:30 hs.

## 1. Enunciado

Se tiene un sistema con sensores de temperatura que realiza mediciones a lo largo de un período de tiempo. Dichas mediciones se almacenaron en un archivo de texto en formato CSV.

Se pide **diseñar** e **implementar** un programa que cargue un archivo de estas características y compute promedios a lo largo de una ventana deslizante de tamaño fijo.

El nombre del archivo de entrada, el nombre del archivo de salida y la cantidad de elementos de la ventana se reciben como parámetros por línea de comandos, en ese orden. Por ejemplo, podría invocarse de la siguiente forma: `python3 programita_taller3.py entrada.csv salida.csv 20`

La línea de comandos anterior indica que se debe cargar el archivo de entrada `entrada.csv`, escribir los promedios en `salida.csv`, y el tamaño de la ventana deslizante es 20.

## 2. Formato de entrada

Cada línea del archivo de entrada tiene los siguientes campos:

`timestamp,temp_1,temp_2,temp_3,temp_4,...,temp_n`

El primer campo de cada fila es un *timestamp*, que se encuentra ordenado crecientemente, y que nos indica la fecha y la hora en la que se realizó la medición. Este campo se encuentra en el formato ISO-8601<sup>1</sup>. Por ejemplo, el *timestamp* que representa la medianoche del 25 de diciembre de 2002 es el siguiente: `2002-12-25T00:00:00`.

Los campos `temp_i` son los valores de temperatura de cada sensor para esa medición, cuyo tamaño no varía dentro del mismo archivo, pero sí puede hacerlo entre distintas entradas. Ocasionalmente algún sensor puede fallar al momento de realizar la medición, en cuyo caso el archivo de entrada tendrá el texto NA en lugar del valor de la temperatura.

---

<sup>1</sup>[http://en.wikipedia.org/wiki/ISO\\_8601](http://en.wikipedia.org/wiki/ISO_8601)

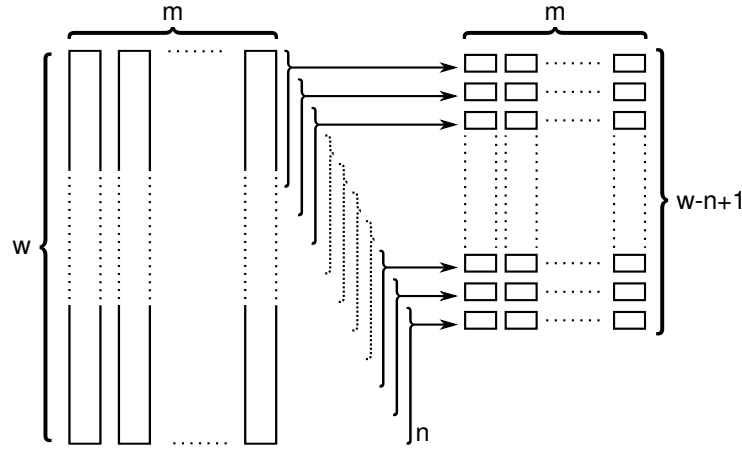


Figura 1: Cálculo del promedio por ventana deslizante

### 3. Cálculo del promedio

El cálculo del promedio se realiza según el procedimiento descrito en la figura 1. En dicha figura, se indica con  $m$  la cantidad de sensores (cantidad de columnas sin contar el timestamp),  $w$  la cantidad de mediciones (cantidad de filas) y  $n$  el tamaño de la ventana deslizante. Los promedios se calculan para cada sensor independientemente, tomando  $n$  mediciones contiguas dentro de la ventana. Si alguno de los valores dentro de esta última fuera NA, el cálculo debe dar NA. En caso contrario, el valor del promedio debe aparecer en la salida como un número con dos cifras decimales significativas. La cantidad de promedios resultantes (es decir, la cantidad de ventanas de tamaño  $n$ ) está determinada por el valor  $w-n+1$ .

### 4. Formato de salida

Cada uno de los promedios debe ser almacenado en el archivo de salida en el orden en el que son calculados, y respetando el siguiente formato:

`rango_de_tiempo,promedio_1,promedio_2,promedio_3,...,promedio_n`

El campo `rango_de_tiempo` indica la cantidad de segundos<sup>2</sup> entre la primera medición y la última de la ventana correspondiente. El archivo de salida debe tener tantas líneas como ventanas deslizantes de  $n$  elementos haya en el archivo de entrada, o ninguna en caso de no contar con suficientes mediciones en el archivo de entrada.

Por ejemplo, si la entrada fuera un archivo con el siguiente contenido:

```
2016-05-03T02:17:24,NA,47.60,39.31,2.69,31.56
2016-05-03T02:17:26,19.13,49.44,54.12,24.34,31.85
2016-05-03T02:19:12,46.12,76.07,31.86,14.37,17.18
2016-05-03T02:20:40,15.67,55.12,61.51,64.71,40.26
```

La salida correspondiente para una ventana deslizante de tamaño 3 sería:

<sup>2</sup>Puede utilizar `datetime.datetime.strptime(timestamp, '%Y-%m-%dT%H:%M:%S')` para interpretar los timestamps en ISO-8601. Luego, si  $t_1$  y  $t_2$  son timestamps ya interpretados, la cantidad de segundos se obtiene como `(t2 - t1).total_seconds()`.

108.0,NA,57.70,41.76,13.80,26.86  
194.0,26.97,60.21,49.16,34.47,29.76

**Condiciones de entrega:**

- Generar un archivo Python con la solución.
- Se evaluará la correctitud del código producido, su claridad y legibilidad y el correcto uso de la herramienta git.
- Agregar la cuenta `icb2016` con permisos de lectura al repositorio `Bitbucket` del alumno. Los docentes descargarán la última versión de los archivos directamente de ahí, luego de ser informados de que el taller se encuentra listo.
- Enviar dicho aviso por correo electrónico a la lista de los docentes de la materia: **`icb-doc@dc.uba.ar`**.
- El mail deberá tener el siguiente *subject*:  
“[Flash 3]: Gonzalez 666/21”.
- En el cuerpo del mail, deben copiar el **comando completo** de clone de manera de que al **copiar y pegar** se pueda bajar la información. Si el comando no funciona, el tp se desaprueba, así que recomendamos fuertemente que lo prueben **antes**.
- Entregar además de forma impresa el código del taller y el *log* del repositorio `Bitbucket` correspondiente.

**Importante:** Solo se admite la entrega por medio de `Bitbucket`.