

## Programación III

### Proyecto Segundo parcial

#### **Introducción:**

Debido al caso de pandemia del año 2020, deseamos hacer un visualizador de datos de los casos de Argentina. Para esto se quiere poder mostrar datos ordenados y agrupados en distintas categorías. Como así también hacer un procesado simple de los datos y mostrar alguna información extra.

#### **Enunciado:**

Realizar una aplicación informática que lea el archivo "CSV" emitido por el Ministerio de Salud con los casos registrados hasta la fecha. El programa será comandado por línea de comandos para realizar determinadas tareas.

El archivo debe ser leído y procesado para brindar la siguiente información según los argumentos que se le pasen.

#### Información estadística:

- Cantidad total de muestras
- Cantidad total de infectados
- Cantidad de fallecidos
- % de infectado por muestras
- % de fallecidos por infectados
- Cantidad de infectados por rango etario ( rango de 10 años)
- Cantidad de muertes por rango etario ( rango de 10 años)

#### **Manual de uso:**

##### **Uso normal:**

covid19.exe [ Argumentos ] Covid19Casos.csv

##### **Argumentos posibles:**

<b>-estad</b>	Mostrará la información estadística.
<b>-p_casos [n]</b>	Mostrará las <b>n</b> primeras provincias con más contagios ordenadas de más a menos. Si <b>n</b> no es pasado, se mostrarán todas las provincias.
<b>-p_muertes [n]</b>	Mostrará las <b>n</b> provincias con más muertes ordenadas de más a menos. Si <b>n</b> no es pasado, se mostrarán todas las provincias.
<b>-casos_edad años</b>	Mostrará los datos de los casos donde la edad sea ' <b>años</b> ' (ordenados por nombre de provincia).
<b>-casos_cui [fecha]</b>	Mostrará los datos de los casos que estuvieron en cuidados intensivos ordenados por fecha de cuidados intensivos, si <b>fecha</b> está indicada, se mostrarán solo las fechas mayores a esta.

**Objetivo:**

Ejercitar y afianzar el uso de estructuras de datos mediante una aplicación concisa y de uso general.  
Afianzar la importancia del uso de las estructuras correctas para mejorar el rendimiento de las aplicaciones.  
Tener en cuenta el impacto en performance introducidos por el uso de estructuras equivocadas o mala implementación en código.

**Formas de presentación:**

**Defensa del trabajo:** Explicación de lo realizado y defensa de la elección de las estructuras de datos.

**Informe del desarrollo:** Informe explicando que estructuras se utilizaron, porque se decidió su utilización. También incluir detalles de pruebas, tareas, complicaciones, retrabajos que hayan ocurrido durante el desarrollo del trabajo.

**Código fuente y comentarios:** Se corregirá el código fuente desde el repositorio de **GitHub** del grupo. El código debe ser prolijo y bien comentado (no sobrecomentar)

**Forma de trabajo:** Grupos de no más de 2 personas.

**Forma de evaluación:** Se evaluará de acuerdo a los siguientes criterios:

Puntualidad en la forma de entrega del trabajo.

Presentación (interfaz, código, prolijidad)

Solución óptima al problema dado – velocidad de ejecución, uso de memoria, etc.

Utilización y adaptación de los códigos proporcionados en clase y del libro.

Conocimiento de la/las estructura/s de datos utilizadas.

Exposición y defensa del trabajo.

Preguntas sobre las estructuras utilizadas a cada integrante.

No se tolerará la deshonestidad académica que comprenda el plagio de código, lo cual implicará la reprobación del parcial de forma irrevocable.

**Cada grupo tendrá 15 minutos para presentar lo desarrollado el día de la entrega.**