

TICS311: Tarea #2

Universidad Adolfo Ibañez

Felipe Aguilera
felipe.aguilera@edu.uai.cl

Danilo Bórquez Paredes
danilo.borquez.p@uai.cl

Sebastián Sáez
sebastian.saez@edu.uai.cl

5 de Abril de 2021

Objetivos

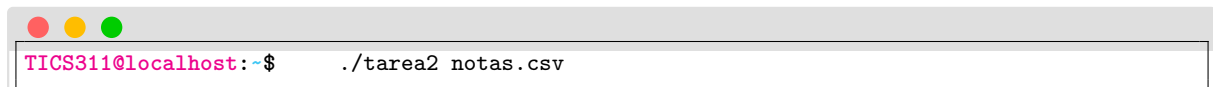
- Internalizar conceptos de ordenamiento
- Trabajo con memoria dinámica

1. Promedios! Pero ordenados

El mismo profesor del trabajo anterior sigue complicado! Luego de su excelente trabajo obteniendo los promedios de sus alumnos, ahora necesita tener la lista **ordenada** de sus alumnos en forma decreciente, según sus promedios de notas. Por esa razón ha acudido a usted, como experto en Programación en C, para obtener los resultados esperados.

1.1. Entrada del programa

Su programa recibirá como entrada la ruta del archivo que contiene todas las notas de los alumnos de un curso, tal como se muestra en el ejemplo siguiente.



```
TICS311@localhost:~$ ./tarea2 notas.csv
```

El nombre del archivo (notas.csv) es sólo un ejemplo, dado que puede llamarse de cualquier manera. El formato de este archivo sí es estricto, y a continuación se muestra un ejemplo:

notas.csv						
nombre,	prueba 1,	prueba 2,	prueba 3,	prueba 4,	proyecto,	tareas
Danilo Borquez Paredes,	1.8,	1.9,	1.1,	5.5,	4.1,	4.5
Felipe Aguilera,	2.9,	4.4,	4.1,	6.9,	5.8,	4.4
Sebastian Saez,	1,	2.5,	5.9,	4.7,	3.5,	1.4

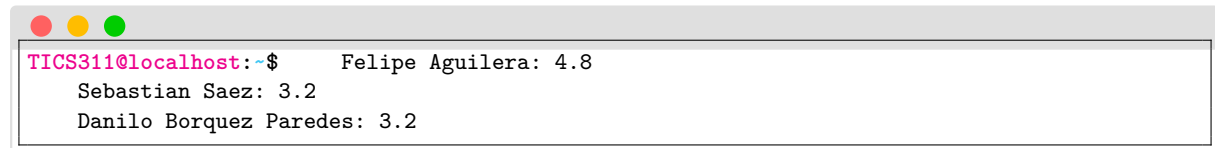
Las características del archivo son las siguientes:

- La cantidad de alumnos es variable, pero siempre hay **a lo menos 1**.
- La primera línea del archivo siempre contiene la información de cada instrumento de evaluación.
- Cada dato siempre estará separado por una coma (,).
- El separador de decimales será el punto (.).

- Nunca habrán campos vacíos (no habrán dos comas seguidas).
- Los instrumentos de evaluación y los nombres de los alumnos pueden tener espacios.
- No habrán caracteres especiales en los nombres como tildes o ñ.

1.2. Salida del programa

El programa debe mostrar por la terminal los alumnos en orden decreciente según sus promedios aritméticos.

A screenshot of a terminal window with a light gray title bar and three colored window control buttons (red, yellow, green) on the left. The terminal text shows a prompt 'TICS311@localhost:~\$' followed by three lines of output: 'Felipe Aguilera: 4.8', 'Sebastian Saez: 3.2', and 'Danilo Borquez Paredes: 3.2'.

```
TICS311@localhost:~$ Felipe Aguilera: 4.8
Sebastian Saez: 3.2
Danilo Borquez Paredes: 3.2
```

2. Sobre la entrega

- La tarea debe ser hecha en lenguaje de Programación C.
- Debe utilizar **MergeSort** o **Quicksort** para el ordenamiento. El no cumplimiento de este requisito es causal de nota máxima 3.0.
- Cada grupo puede ser de 2 o 3 personas.
- La tarea se debe entregar el día **Viernes 23 de Abril** a las 23:59.
- Por cada día de atraso se descontará 1 punto, comenzando a las 00:00 horas del siguiente día. Por ejemplo si entrega la tarea a las 00:00 del siguiente día, la nota máxima que puede obtener es un 6.0
- Para la corrección se utilizará un compilador gcc v 5.1 o superior
- La entrega se realiza por la plataforma Webcursos¹
- El archivo a entregar debe ser un zip que contenga una carpeta en su interior (y sólo una carpeta) con el nombre **tarea2**. Dentro de esa carpeta debe haber un Makefile² y por lo menos un main.c. Además el nombre de su zip debe ser grupoX-tarea2.zip, donde X es el número de su grupo.
- **IMPORTANTE:** el directorio **tarea2** debe contener el Makefile que generará el archivo ejecutable **tarea2**.

¹<http://webcursos.uai.cl>

²Este archivo deben generarlo. Información útil pueden encontrarla en <https://stackoverflow.com/questions/1484817/how-do-i-make-a-simple-makefile-for-gcc-on-linux>. Pueden probar su Makefile en un computador con MAC o Linux, o en VSCode en la terminal con UBUNTU.