



ALUMNO:

EPS

Asignatura: G0460004 Programación I

Curso: 2021/2022

Semestre: 1º

Examen: Final

Convocatoria: Extraordinaria

Fecha: 20-06-2022

## Parte Práctica (10 puntos; 70% nota final)

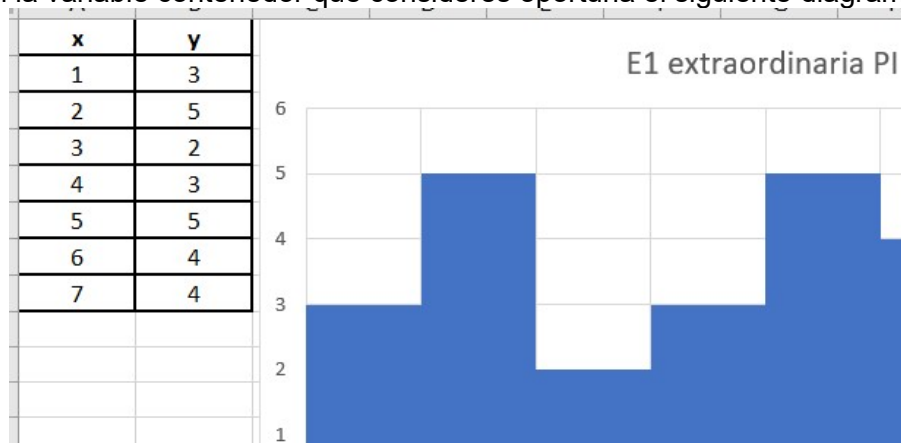
Se debe entregar los ficheros de vuestros ejercicios mediante la actividad en el campus antes de la finalización del mismo.

Tiempo: 2 horas 30 minutos.

## Ejercicio 1 [3 puntos]

*Se debe entregar un único archivo ejercicio1.cpp*

Representar con la variable contenedor que consideres oportuna el siguiente diagrama de escalera.



Debes implementar:

- Una función que devuelva el rango del conjunto de datos (menor y mayor valor) [0,75 puntos]
- Una función que devuelva la moda del conjunto de datos (el que más se repita). Teniendo en cuenta que puede haber moda multimodal, es decir, que haya varios valores con el número máximo de repeticiones. [1.5 puntos]
- Una función que devuelva la mediana de los valores de la gráfica sin repetición (con la secuencia ordenada de menor a mayor la mediana de un conjunto impar de elementos es el que ocupa la posición central. Si el grupo de elementos es par la mediana será la media de los 2 valores centrales [0,75 puntos]

**Para puntuar los apartados de este ejercicio debes probar las funciones implementadas en la función principal del programa.**



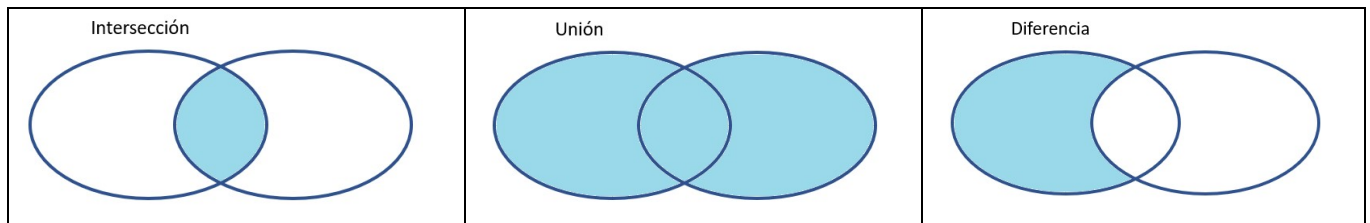
## Ejercicio 2 [3 puntos]

*Se debe entregar un único archivo ejercicio2.cpp*

Vamos a representar dos conjuntos de letras como dos variables de tipo `std::string` {"arboles"}, {"arboleda"}.

Debéis implementar las siguientes funciones que recibirán como argumentos los dos `std::string` indicados:

- Intersección: dicha función devolverá un `std::string` con los caracteres comunes a ambas palabras sin repetirse ninguno. [0.75 puntos]
- Unión: dicha función devolverá un `std::string` con los caracteres de ambas palabras sin repetirse ninguno. [0.75 puntos]
- Diferencia: dicha función devolverá los caracteres del `std::string` del primer parámetro que no estén en el segundo sin repetirse ninguno. [1.5 puntos]



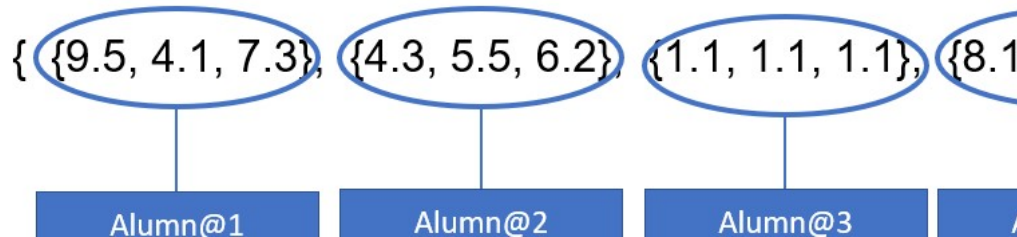
**Para puntuar los apartados de este ejercicio debes probar las funciones implementadas en la función principal del programa.**



## Ejercicio 3 [4 puntos]

*Se debe entregar un único archivo ejercicio3.cpp*

Para almacenar las calificaciones de una asignatura durante un curso se utiliza una variable *Notas* que contiene las calificaciones de cada uno de los alumnos de la forma que se indica en la imagen.



Dicha variable contiene las siguientes calificaciones:

{ {9.5, 4.1, 7.3}, {4.3, 5.5, 6.2}, {1.1, 1.1, 1.1}, {8.1, 10, 9.5} }

Se pide implementar las siguientes funciones:

- Media: función lambda que devolverá la media de los valores contenidos en un vector<double> [0.25 puntos]
- Máximo: función lambda que devolverá el máximo de los valores contenidos en un vector <double> [0.25 puntos]
- Mínimo: función lambda que devolverá el mínimo de los valores contenidos en un vector <double> [0.25 puntos]
- Función composición que reciba por argumento la variable *Notas* y la función lambda que corresponda. Dicha función devolverá un vector<double> con los resultados de la operación solicitada [0.75 puntos]
- Función composición que reciba por parámetro de entrada la variable *Notas* y 2 funciones lambda. De manera que permita realizar operaciones múltiples sobre la variable. Como por ejemplo calcular el máximo de las medias. [2.5 puntos]

**Para puntuar los apartados de este ejercicio debes probar las funciones en la función principal del programa.**

## Rúbrica de evaluación

|  |      |
|--|------|
| El programa no compila o no se asemeja a lo que se pide  | 0%   |
| El programa no se aproxima lo suficiente a lo pedido   | 40%  |
| El programa se aproxima suficientemente a lo pedido  | 60%  |
| El programa funciona correctamente y las estrategias y elementos de código elegidos son adecuados. | 100% |