**Nombre y Apellido: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Cantidad de hojas entregadas: \_\_ Nota:\_\_\_\_\_\_\_\_**

**Tema A**

|  |
| --- |
| **Lea atentamente TODO el enunciado antes de empezar a desarrollar. Tener en cuenta casos borde en todos los ejercicios**  **Condición necesaria (y suficiente) de aprobación 6 puntos con al menos 1 test del punto 3 en OK** |

1) Se pide implementar una función validar(nombre,apellido, password) que recibe un nombre, un apellido y una password.

El mismo debe validar que la password sea válida. Devolviendo True si la contraseña es válida, False si no lo es.

Una password es válida si cumple lo siguiente:

1. Debe tener entre 8 y 15 caracteres
2. La contraseña no puede contener el nombre ni el apellido ya sea en minúscula como en mayúscula
3. Contiene al menos uno de los siguientes símbolos: % $ - ( ) = # ! ?

2) Un grupo de amigos quieren juntarse a cenar pero tienen todos calendarios muy ajustados y les es difícil ver qué días pueden juntarse todos. Se pide escribir una función organizar(dic) que reciba un diccionario con los nombres de cada persona, y que tengan como valor asociado los días en los que esa persona no puede juntarse (escritos del 1 al 30, en forma de lista), y que devuelva una lista con los días del mes (escritos del 1 al 30) en los que todos pueden juntarse.

3) Se cuenta con una implementación de lista enlazada con únicamente una referencia al primer

nodo. Se pide implementar un método de ListaEnlazada llamado `invertir\_lista` que invierta

el orden de la lista. (No crea una lista nueva, sino que invierte a la misma lista)

Ejemplo:

L = 1 -> 5 -> 9 -> 9 -> 2 -> 9

L.invertir\_lista => L = 9 -> 2 -> 9 -> 9 -> 5 -> 1