



UNIVERSIDAD TECNOLÓGICA NACIONAL

FACULTAD REGIONAL ROSARIO

TRABAJO PRÁCTICO Nº 3: CONSIGNAS

ALGORITMOS Y ESTRUCTURAS DE DATOS

INTRODUCCIÓN	3
DESARROLLO	3
Módulo 0: inicialización	5
Módulo 1: Estudiantes	5
Módulo 2: Moderadores	7
Módulo 3: Administradores	8
BONUS TRACK 1: PUNTUANDO CANDIDATOS	9
BONUS TRACK 2: SUPER-LIKE	9
BONUS TRACK 3: REVELAR CANDIDATO	9

Introducción

En este TP3 vamos a introducir el concepto de **registros** y **archivos**.

Deben realizar la codificación en lenguaje Python de lo requerido para este TP3.

Junto al código de Python, indicar claramente la **declarativa de variables** utilizadas, en especial, las referidas a las variables del tipo registros, archivos y arrays de registros.

Tener en cuenta al implementar los archivos en Python, respetar el marco teórico de la cátedra sobre archivos, es decir, se debe utilizar la **librería pickle**, y los archivos deben contener registros del mismo tamaño dentro de cada fila.

Antes de comenzar con este TP, por favor leer el documento **Enunciado General** de manera completa, para saber de qué va el negocio, y entender el modelo de datos requerido. Es obligatoria también la lectura de las **preguntas frecuentes**, para evitar prácticas que resten puntos al momento de la entrega.

Desarrollo

En esta tercera etapa deberán realizar lo siguiente:

- Al comenzar con la ejecución del programa, hacer que el usuario elija la opción “logueo” o “registrarse”.
 - Los estudiantes, moderadores y administradores deberán guardarse en archivos. Sólo los estudiantes se pueden dar de alta al comienzo de la ejecución del programa. Los administradores deberán estar previamente cargados en el archivo, y no podrán darse de alta en la sección de “registro”. Los moderadores sólo podrán darse de alta a través de un administrador (ver sección 1.b del menú de administradores).
 - Para acceder a la sección de logueo, deberá haber aunque sea 1 moderador, 1 administrador y 4 estudiantes cargados.
 - Reutilizar la mayor cantidad del código posible del TP2, a menos que explícitamente se haya pedido reformularlo en las correcciones. Obviamente, adaptarlo para que en lugar de utilizar arreglos utilice archivos.
 - No puede haber 2 usuarios que compartan el mismo email. Chequear también que no existan usuarios de distinto tipo con el mismo email.
- Hacer el menú para cada tipo de usuario (Estudiante, Administrador y Moderador).
 - Pedir un nombre de usuario (email) y contraseña al operador. Verificar que existan el usuario y su clave guardados en el sistema, y de acuerdo al tipo de usuario que está ingresando, presentarle el menú correspondiente.

- Tener en cuenta que, además de ingresar un par usuario / contraseña correctos, también se debe chequear que el estado del usuario sea “ACTIVO” (string). Caso contrario, el login NO será correcto.
- Tanto el ID de cada estudiante como de cada moderador o administrador va a ser siempre un número entero auto-incremental, que comienza en 0.

Si el usuario que ingresa (por medio de su nombre de usuario y su clave) coincide con un usuario guardado, y el mismo es del tipo **Estudiante**, deberán mostrar el menú completo correspondiente a un usuario de ese tipo:

1. Gestionar mi perfil
 - a. Editar mis datos personales
 - b. Eliminar mi perfil
 - c. Volver
2. Gestionar candidatos
 - a. Ver candidatos
 - b. Reportar un candidato
 - c. Volver
3. Matcheos
 - a. Ver matcheos
 - b. Eliminar un matcheo
 - c. Volver
4. Reportes estadísticos
0. Salir

Tener en cuenta que apenas se ejecuta el programa, solamente se verán las opciones:

1. Gestionar mi perfil
2. Gestionar candidatos
3. Matcheos
4. Reportes estadísticos
0. Salir

Y luego las opciones con letras o sub-opciones se mostrarán luego de elegir alguna opción principal. Al elegir la opción “Volver” en cualquiera de los sub-menús, se volverá al menú anterior.

Si el usuario que ingresa (por medio de su nombre de usuario y su clave) coincide con un usuario guardado, y el mismo es del tipo **Moderador**, deberán mostrar el menú completo correspondiente a un usuario de ese tipo:

1. Gestionar usuarios
 - a. Desactivar usuario

- b. Volver
- 2. Gestionar Reportes
 - a. Ver reportes
 - b. Volver
- 3. Reportes Estadísticos

Si el usuario que ingresa (por medio de su nombre de usuario y su clave) coincide con un usuario guardado, y el mismo es del tipo **Administrador**, deberán mostrar el menú completo correspondiente a un usuario de ese tipo:

- 1. Gestionar usuarios
 - a. Eliminar un usuario (incluyendo moderadores)
 - b. Dar de alta un moderador
 - c. Desactivar usuario
 - d. Volver
- 2. Gestionar Reportes
 - a. Ver reportes
 - b. Volver
- 3. Reportes Estadísticos

Módulo 0: inicialización

Si no existen likes cargados, desarrollar un procedimiento que se llame al comienzo de la ejecución de nuestro programa. El mismo deberá recorrer el archivo de likes, y llenarla de 0s y 1s de manera aleatoria. De esta manera, cada vez que el programa se ejecute, vamos a tener cargada cierta interacción entre los estudiantes. Si ya se tienen likes cargados en el archivo correspondiente, NO llamar a dicho procedimiento.

Módulo 1: Estudiantes

Para esta tercera etapa, se pedirán realizar los módulos:

- 1. Gestionar mi perfil
 - a. Editar mis datos personales
 - b. Eliminar mi perfil
- 2. Gestionar candidatos

- a. Ver candidatos
- b. Reportar un candidato
- 4. Reportes Estadísticos

El resto de las opciones deberán mostrar un cartel aclaratorio que diga “En Construcción”.

Descripción de los módulos:

1.a Editar mis datos personales: en esta opción, el estudiante logueado deberá poder modificar la información ingresada para su fecha de nacimiento, biografía, hobbies y cualquier otro campo que crean relevante.

1.b Eliminar mi perfil: mostrar un cartel aclaratorio, preguntando confirmación si realmente se desea eliminar el perfil. En caso de elegir la opción “sí”, cambiar el estado del usuario por inactivo, y volver a la pantalla de login del trabajo.

2.a Ver candidatos: al ingresar a esta opción, se deben mostrar todos los estudiantes. Mostrar nombre, fecha de nacimiento, edad, biografía y hobbies. Una vez mostrada toda la información, permitir ingresar en una variable me-gusta el nombre del estudiante con el cual le gustaría en un futuro hacer un Matcheo. Verificar que el nombre sea correcto.
Para mostrar la edad, sabiendo que la fecha de nacimiento se guarda como string en el formato YYYY-MM-DD siendo YYYY el año, MM el mes y DD el día (ej: el 1 de diciembre de 2002 sería 2002-12-01), calcular y mostrar la edad del candidato.

Se recomienda guardar en un **archivo un tipo de registro Like**, donde se guarde el Id del remitente del like (quién lo dio) y el Id del destinatario (a quién). Codificar dos funciones auxiliares, una para saber si una persona le dio like a otra (usando los IDs de ambos como parámetros), y otra para saber si dos personas tienen likes correspondidos (también usando ambos IDs como parámetros=).

	remitente	destinatario
0	0	1
1	0	2
2	0	3
3	1	4
4	2	3
5	3	2
6	4	3
7	3	4

En este ejemplo: el Id 0 le ha dado me gusta a 3 estudiantes, al 1, al 2 y al 3. El Id 1 solamente le ha dado me gusta al id 4. Esto quiere decir que el usuario 0 le dió like al usuario 1, pero el usuario

1 solo le dió like al usuario 4, por ende, No hay match. Mientras que el id 2 le dio like al 3 y el 3 al 2 por ende, Si hay match.

2.b Reportar un Candidato: al ingresar a esta sección, se deberá pedir el ID o nombre de usuario (cualquiera de los dos) de un candidato, y el motivo por el cual se está reportando al mismo. Validar que el candidato exista, y dar de alta un reporte con el ID del usuario actual como reportante, el ID del usuario reportado, el motivo dado y el estado inicial 0 (*ver sección 2.a de moderadores para obtener más información acerca de los distintos estados de un reporte*).

4. Reportes Estadísticos: mostrar con qué porcentaje de todos los candidatos posibles (distintos a nosotros) se matcheo ambas veces, a cuántos le pusimos “me gusta” pero no nos han devuelto el match, y cuántos estudiantes nos dieron “me gusta”, a los cuáles nosotros no le hemos dado “me gusta todavía”. Mostrarlo de la siguiente forma:

Matcheados sobre el % posible: 50%

Likes dados y no recibidos: 2

Likes recibidos y no respondidos: 3

Módulo 2: Moderadores

Para esta tercera etapa, se pedirán realizar los módulos:

1. Gestionar usuarios
 - a. Desactivar usuario
 - b. Volver
2. Gestionar Reportes
 - a. Ver reportes
 - b. Volver

1.a Desactivar un Usuario: se ingresa el ID o el nombre de usuario, se busca si el usuario existe, y luego se muestra una pantalla idéntica a la de “Eliminar mi Perfil”.

2.a Ver reportes: los reportes pueden tener como estado 0, 1 y 2. 0 quiere decir que el reporte todavía no ha sido visto por un moderador o un administrador, 1 quiere decir que el reporte ha sido tomado y el reportado ha sido desactivado, y 2 quiere decir que el reporte ha sido ignorado.

Se mostrarán los reportes los cuáles tanto el usuario reportante como el reportado están activos, y que tengan un estado = 0. Por cada reporte, se preguntará cómo se quiere proceder:

ignorar reporte o bloquear al reportante. En caso de seleccionar la primer opción, el estado del reporte será actualizado a 2. En caso de seleccionar la segunda opción, el estado del reporte será actualizado a 1, y el usuario reportado será desactivado (su estado será inactivo).

Módulo 3: Administradores

Además de todo lo referido a moderadores, se pedirá en esta tercera etapa agregar las secciones descritas en negrita:

1. Gestionar usuarios
 - a. Eliminar un usuario (incluyendo moderadores)**
 - b. Dar de alta un moderador**
 - c. Desactivar usuario
 - d. Volver
2. Gestionar Reportes
 - a. Ver reportes
 - b. Volver
- 3. Reportes Estadísticos**

1.a Eliminar un usuario: se desea eliminar del archivo (borrado físico) a un usuario, ya sea estudiante o moderador. Un administrador NO podrá borrar otros administradores.

1.b Dar de alta un moderador: esta sección será similar a la de “registro”, con la diferencia que sólo los administradores podrán crear desde su menú (una vez logueados) nuevos moderadores.

3 Reportes estadísticos: además de las mismas métricas que ve un moderador, se desean obtener:

- La cantidad de reportes realizados por los estudiantes
- El porcentaje de reportes ignorados
- El porcentaje de reportes aceptados (donde se ha bloqueado al estudiante)
- El moderador que mayor cantidad de reportes ha ignorado
- El moderador que mayor cantidad de reportes ha aceptado
- El moderador que mayor cantidad de reportes ha procesado (ignorados + aceptados)

Bonus Track 1: puntuando candidatos

Se desea realizar un algoritmo que nos permita darle un “puntaje” a los candidatos, de acuerdo a la cantidad de likes recibidos y dados. Vamos a llamar like “correspondido” a un match, en otras palabras, si le damos like a otra persona, y esa persona nos dio like a nosotros va a ser un like “correspondido”.

Se desea recorrer secuencialmente el archivo de likes. Si encontramos un like dado por el candidato y correspondido por la otra persona, sumarle 1 punto. Si encontramos 1 like que fue dado y no correspondido, restarle 1 punto.

Además, se desea dar 1 punto más si la “racha” de likes correspondida es mayor o igual a 3. Esto es, a partir del 3er like correspondido sumar 1 punto extra siempre y cuando la racha se mantenga mayor o igual a 3. Si volvemos a encontrar un like no correspondido la racha vuelve a 0.

Emitir un listado de candidatos según su puntaje, que sólo podrán ver los administradores.

Bonus Track 2: super-like

Al registrarse un nuevo estudiante, se le podrá asignar 1 “super-like”. Al ver los candidatos, el usuario podrá utilizarlo con cualquier persona. Al dar un “super like”, automáticamente el like que dé, va a ser correspondido (aunque la otra persona no haya dado like, automáticamente se realizará el match).

En otras palabras, completar acá automáticamente 2 registros para que haya matcheo, o sea, 1 registro con el like que dé el estudiante que tenga el 'super-like', y el otro registro para que tenga su correspondencia automática, es decir, ida y vuelta. Como si al que le diste like te devuelve el like automáticamente.

Bonus Track 3: revelar candidato

Al registrarse un nuevo estudiante, se le otorgarán 1 crédito para “revelar candidatos”. Si el estudiante aún no ha utilizado dicho crédito, se mostrará una nueva opción en el menú, justo debajo de “ver candidatos”, que mostrará hasta 3 personas que le han dado like al estudiante logueado (y que no hayan sido correspondidos aún por este).

Modelo de Datos

estudiantes

id_estudiante: entero
email: string[32]
nombre: string[32]
sexo: char
contraseña: string[32]
estado (activo / inactivo): booleano
hobbies: string[255]
materia_favorita: string[16]
deporte_favorito: string[16]
materia_fuerte: string[16]
materia_debil: string[16]
biografia: string[255]
país: string[32]
ciudad: string[32]
fecha_nacimiento: string[10]

moderadores

id_moderador: entero
email: string[32]
contraseña: string[32]
estado: booleano

administradores

id_admin: entero
email: string[32]
contraseña: string[32]

likes

remitente: entero
destinatario: entero

reportes

id_reportante: entero
id_reportado: entero
razón_reporte: string[255]
estado: entero