

1.a)	20 puntos		
def prod_cli(rut, movimiento):			
mov_banco = open(movimiento)			Observación: Una línea está correcta si está en la posición correcta (funcionalmente) y con la indentación correcta.
sumaprod = 0	2 ptos		
saldo = 0			
c = set()			
for li in mov_banco:	1 pto		
r,p,s,_ = li.strip().split(',')	1 pto		
if r == rut:	1 pto		
if p not in c:	1 pto		
sumaprod += 1	2 ptos		
c.add(p)			
saldo = saldo + int(s)	1 pto		
mov_banco.close()	1 pto		
return sumaprod, saldo	1 pto		
1.b)			
def archivo_prod(cli, movimiento):			
cli_banco=open(cli)	1 pto		
prod=open('productos.txt','w')			
f = '{0}/{1}/{2}/{3}\n'	1 pto		
for li in cli_banco:	1 pto		
li = li.strip().split('#')	1 pto		
r,n,_ = li	1 pto		
cant,s = prod_cli(r,movimiento)	1 pto		
prod.write(f.format(r,n,cant,s))	1 pto		
prod.close()	2 ptos		
cli_banco.close()			

Pregunta 2:		40 ptos			
Item	0 Pto	4 Ptos	7 Ptos	10 Ptos	13 Ptos
Manejo de archivos	No demuestra tener noción de cómo usar archivos o no cierra los archivos abiertos en al menos la mitad de los casos.	Al menos en una función utiliza de forma correcta archivos. Cierra los archivos de forma incorrecta o abre los archivos con el nombre de los archivos del ejemplo y no según el parámetro de la función.	Usa correctamente los archivos en todos los casos, abre y cierra donde corresponde, utiliza el nombre recibido como parámetro y no el nombre de los ejemplos.		
Pregunta a)	No demuestra tener noción de cómo resolver el problema	Resuelve para algunos casos solamente, olvida algún detalle. Ej: no agrega el .txt al nombre del archivo, no agrega un 0 a los días o meses menores a 9, no pasa los valores numéricos a str si es necesario	Retorna correctamente el resultado esperado		
Pregunta b)	No demuestra tener noción de cómo resolver el problema	Demuestra tener una noción de qué hacer, tiene la lógica de la solución pero con varios errores en el proceso. Ej: no logra obtener el usuario de la línea, calcula el rango de los mensajes independiente del usuario.	Comete algún error importante o 2 errores leves. Ej: no retorna (24, 24) si el usuario no envió mensajes, no considera el caso en que el usuario sólo envió un mensaje, problemas leves con el split.	Comete un error leve. Ej: olvida pasar a entero el valor de la hora, olvida las comillas al caracter divisor en el split.	Retorna correctamente el resultado esperado
Pregunta c)	No demuestra tener noción de cómo resolver el problema	Demuestra tener una noción de qué hacer, tiene la lógica de la solución pero con varios errores en el proceso. Ej: utiliza de forma incorrecta alguna estructura de datos para contar los mensajes por usuario, no inicializa el contador en 0.	Comete algún error importante o 2 errores leves. Ej: retorna una lista ordenada de menor a mayor o no por la cantidad de mensajes enviados, entrega la lista completa sin utilizar el parámetro N, problemas con el comando split o intenta aplicar split sobre split, cuenta los mensajes donde apareció mencionado el usuario.	Comete un error leve. Ej: no inicializa la estructura de datos usada, no inicializa en 0 el contador, olvida las comillas al caracter divisor en el split, hace un append de dos valores y no una tupla.	Retorna correctamente el resultado esperado
Descuentos		0 Pto			
Forma y expresiones	No comete errores graves con respecto al lenguaje; a lo más errores atribuibles más a descuidos que ignorancia (por ejemplo, en una ocasión olvidar los dos puntos al final de la línea, confunde = y == o confunde () y [] o , ocupa ≤ en lugar de <=, pero el resto de las veces lo hace bien)		Demuestra no entender construcciones importantes del lenguaje (por ejemplo: no indenta, le pone una condición al else, confunde el string x con la variable x, asigna valor a variables de forma incorrecta 4=x), consistentemente no le pone los paréntesis finales a d.items(), d.values(), línea.strip() o arch.close(). Un descuento de -4 por tipo de error.		

Pregunta 3: 40 pts					
Item	0 Pto	4 Ptos	7 Ptos	10 Ptos	13 Ptos
Manejo de archivos	No demuestra tener noción de cómo usar archivos o no cierra los archivos abiertos en al menos la mitad de los casos.	Al menos en una función utiliza de forma correcta archivos. Cierra los archivos de forma incorrecta o abre los archivos con el nombre de los archivos del ejemplo y no según el parámetro de la función.	Usa correctamente los archivos en todos los casos, abre y cierra donde corresponde, buen uso de los modos de escritura y actualización en todas las preguntas.		
Problema a)	No demuestra tener noción de cómo resolver el problema o los errores son mayores a los aciertos.	La lógica de la respuesta está correcta pero comete algún error en la implementación. Ej: el formato de la línea no es el correcto, olvida el \n al escribir la nueva línea o agrega \n al principio y al final de la línea.	Resuelve correctamente.		
Problema b)	No demuestra tener noción de cómo resolver el problema o los errores son mayores a los aciertos.	Desmuestra tener una noción de qué hacer, tiene la lógica de la solución pero con varios errores en el proceso. Ej: intenta leer y escribir en un archivo al mismo tiempo, hace split sobre split.	La lógica de la respuesta está correcta pero comete algunos errores menores o uno importante. Ej: mal uso del split o intenta aplicar split sobre split, la línea censurada no tiene el formato de los archivos, censura todo no sólo el mensaje, censura palabras que tengan contenida la palabra a censurar.	La lógica de la respuesta está correcta pero comete algún error menor. Ej: no agrega el '\n' al escribir una línea, olvida comparar las palabras en minúscula, no hace strip() de la línea, censura con una @ y no la cantidad según el largo de la palabra.	Resuelve correctamente.
Problema c)	No demuestra tener noción de cómo resolver el problema o los errores son mayores a los aciertos.	Desmuestra tener una noción de qué hacer, tiene la lógica de la solución pero con varios errores en el proceso. Ej: hace split sobre split, no ordena la lista de fechas para que los mensajes aparezcan en orden de emisión, el nombre del archivo creado es igual al del ejemplo y no en base al usuario recibido como parámetro.	La lógica de la respuesta está correcta pero comete algunos errores menores o uno importante. Ej: el nombre del archivo creado no es el especificado, no filtra por el usuario indicado.	La lógica de la respuesta está correcta pero comete algún error menor. Ej: agrega un '\n' extra a la línea, crea el archivo abriéndolo en modo 'a' y no 'w'.	Resuelve correctamente.
Descuentos		0 Pto	-4 Puntos		
Sintaxis y expresiones	No comete errores graves con respecto al lenguaje; a lo más errores atribuibles más a descuidos que ignorancia (por ejemplo, en una ocasión olvidar los dos puntos al final de la línea, confunde = y == o confunde () y [] o , ocupa ≤ en lugar de <=, pero el resto de las veces lo hace bien)		Demuestra no entender construcciones importantes del lenguaje (por ejemplo: no indenta, le pone una condición al else, confunde el string x con la variable x, asigna valor a variables de forma incorrecta 4=x), consistentemente no le pone los paréntesis finales a d.items(), d.values(), línea.strip() o arch.close(). Un descuento de -4 por tipo de error.		