



Apellido y Nombre

DNI: _

Calificación :

Ejercicio 1: El banco "CBCI", mantiene el estado de las cuentas de sus clientes en el archivo cuentas.dat (ordenado por el nro. de cuenta), que actualiza todos los días con los movimientos provenientes del archivo de texto movimientos.txt (ordenado por fecha y hora del movimiento) de la siguiente manera:

- Si el movimiento es un débito (D): se le debe restar al saldo de la cuenta cliente el importe del movimiento.
- Si el movimiento es un crédito (C): se debe sumar el importe del movimiento al saldo de la cuenta cliente.

El archivo de texto de movimientos tiene registros de longitud fija, con 8 caracteres para el código de cuenta, un carácter para el tipo de movimiento y 9 caracteres (con 2 decimales) para el importe del movimiento. Por razones de seguridad el archivo de movimientos está encriptado.

Además, el sistema debe guardar en un archivo de texto llamado igual que el de movimientos pero con el sufijo "_descubierto.txt", todos los movimientos y por orden de aparición que luego de su aplicación en la cuenta, esta quede con saldo al descubierto.

Se pide:

- Realizar la actualización del archivo de cuentas con la información de los movimientos. Debe minimizar el acceso a disco.
- Resolver la creación del archivo de movimientos al descubierto. Para la resolución debe valerse de una cola con implementación estática. Incluir además la resolución de las primitivas del TDA cola.
- Realizar la función de desencriptado de los registros del archivo movimientos teniendo en cuenta que no puede utilizar funciones de biblioteca y debe usar notación y aritmética de punteros. A continuación se especifica cómo fueron encriptados los registros:

Algoritmo de encriptado:

1. Invertir la cantidad de caracteres que indique el dígito en la posición 0 a partir del siguiente carácter.
2. El dígito "n" en la última posición de la cadena indica invertir los "n" caracteres anteriores al mismo.
3. Por último, se encripta cada carácter de la cadena obtenida en el paso anterior utilizando el carácter correspondiente por la posición relativa en la "palabraClave". Si la cadena a encriptar es más larga que la palabraClave, esta última se comprta de manera circular, es decir que se deberá volver al primer carácter de la misma cada vez que esta se termine. Los caracteres se encriptan sumando la posición relativa en el alfabeto de la letra correspondiente de la palabra clave, teniendo en cuenta que los caracteres especiales no se encriptan, y los datos de las claves pueden ser dígitos del 0 al 9, letras mayúsculas de "A" a "Z" o minúsculas de "a" a "z" y que después de "9" viene "A", después de "Z" viene "a" y después de "z" viene "0".

Ejemplo:

PalabraClave: "clave"

Registro: 9875463D004582.99

1. **9875463D004582.99** → **900D3645784582.99** (cadena marcada en negrita invertida)

2. **900D3645784582.99** → **900D3649.28548759** (cadena marcada en negrita invertida)

3.

Cadena	9	0	0	D	3	6	4	9	.	2	8	5	4	8	7	5	9
ValorCadena	9	0	0	13	3	6	4	9	-	2	8	5	4	8	7	5	9
PalabraClave	c	l	a	v	e	c	l	a	v	e	c	l	a	v	e	c	l
ValorClave	38	47	36	57	40	38	47	36	57	40	38	47	36	57	40	38	47
Resultado	l	l	a	8	h	i	p	j	.	g	k	q	e	3	l	h	u

Ejercicio 2: Escribir una clase Complejo cuyos atributos son *real* e *imaginario* (enteros). Declare y desarrolle el constructor parametrizado, con parámetros con valor predeterminado, y la sobrecarga de los operadores necesarios para compilar la función main dada en el proyecto.

Nota:

Suma

$$(a, b) + (c, d) = (a+c, b+d)$$

Producto por escalar

$$r * (a, b) = (r*a, r*b)$$

Multiplicación

$$(a, b) * (c, d) = (a*c - b*d, a*d + b*c)$$

NOTA:

La resolución es inválida en cualquiera de los ejercicios si supone y/o utiliza variables globales.

Crear una carpeta con su Apellido_Nombre, guardar en ella todo lo realizado.

Crear un proyecto para cada ejercicio.

Entregar la carpeta compactada Apellido_Nombre.zip o .rar