

Ejercitación Complementaria
C++ - guía 1
Continuación de la clase cadena.

La siguiente ejercitación tiene como objetivo:

- Afianzar los conceptos básicos del lenguaje C++

Se toma código base la implementación realizada en el aula de la clase cadena.

Ejercicio 1:

Implementar la sobrecarga de comparaciones mayor igual y menor igual.
Revisar si estos métodos deben o no ser *friend*.

Ejercicio 2:

Ej. 2 - Parte 1:

Modificar las sobrecargas de comparación para que no sean *case sensitive*. Es decir que tome las mayúsculas igual que las minúsculas.

Ej. 2 - Parte 2:

Agregar al objeto un método que permita configurar (set) si las comparaciones serán *case sensitive* o no.
Incluir otro método que permita conocer esta configuración (get).

Ej. 2 - Parte 3:

Modificar el ejercicio anterior para que la configuración sea a nivel de la clase y no del objeto.

Ej. 2 - Parte 4:

Basado en los ejercicios anteriores, implementar un esquema que por omisión las comparaciones del objeto sean como indica la clase, y si el objeto es configurado, el comportamiento de estas comparaciones dependa de la configuración del objeto y no de la clase.

Debe contemplar un método para que el objeto vuelva a comportarse según la configuración de la clase.

Ejercicio 3:

Implementar un método que permita buscar un texto dentro de la clase.

El método debe, en caso de encontrar dicho texto, retornar su posición.

Debe permitir:

- que la búsqueda inicie en una posición particular y el sentido de la búsqueda (hacia adelante o hacia atrás).
- contemplar que el texto buscado sea o no case sensitive.

Nota: el texto siempre en el sentido normal.

Ejercicio 4:

Implementar dos opciones para reemplazar un carácter ubicado en una posición determinada.

Opción 1: mediante el método setCaracter

Opción 2: mediante la sobrecarga de los corchetes (operator[])

Notas:

- Como deberían comportarse estos métodos si en la ubicación seleccionada no hay texto o no está reservado.
- ¿Qué diferencias hay en la sobrecarga del operador [], con el implementado en el ejercicio 9 de la serie A de los ejercicios de clase? ¿Pueden coexistir las dos definiciones? ¿En que caso se usa cada una?

Ejercicio 5:

Modificar la sobrecarga de salida para que la información que se imprima sea configurable.

Estas maneras son:

- Tal cual la información esta almacenada.
- Todo en mayúscula.
- Todo en minúscula.
- La primera letra en mayúscula y el resto en minúscula.
- La primera letra de cada palabra en mayúscula y el resto en minúscula.
- Modo debug. Esto es el texto como está, todos los valores internos del objeto y de la clase, y la posición en memoria que ocupar el objeto.

La configuración debe ser a nivel clase y objeto, teniendo precedencia la configuración del objeto por sobre la de la clase.

Ejercicio 6:

Implementar un método que permita reemplazar un texto buscado.

Ejercicio 7:

Modificar la clase para que si al disminuir la cantidad de bytes requeridos por el dato (en unidades de SZ_BLOCKS), es menor al 20% de la cantidad reservada, se reajuste el espacio de memoria utilizado.

Ejercicio 8:

Implementar lo necesario para poder evaluar un objeto en forma directa, sin hacer una comparación explícita. Tanto en su forma afirmativo como negativa.

Por ejemplo:

```
if (c1) xxxxxx;           if (!c1) xxxxxx;
else xxxxxxxx;           else xxxxxxxx;
```

En donde la evaluación afirmativa retorne Verdadero si el parámetro **nchar** es mayor a cero, y falso en caso que sea 0. Y la evaluación negativa funcione a la inversa.