

Normas para la realización del examen:

Duración: 2.15 horas

- Desarrolle el examen en un fichero cpp (así puede aprovechar el resaltado de sintaxis y la indentación). Incluya al principio su DNI, Apellidos y Nombre
- El nombre del fichero debe construirse con sus apellidos y luego el nombre.
- Indique claramente donde comienza cada ejercicio/apartado, utilizando una línea de asteriscos.
- Recuerde que no está escribiendo programas completos y, por tanto, no debe preocuparse de la compilación.
- Todos los ficheros recibidos serán procesados por un programa para la detección de copias.

◁ Ejercicio 1 ▷ Palabras Palisimétricas**[5 puntos]**

Una palabra se define como *palisimétrica* si, cuando se parte por la mitad, da como resultado dos palabras que tienen las mismas letras con las mismas frecuencias. Si la longitud de la palabra es un número impar, la decisión se toma sin considerar la letra central. Por ejemplo gaga es *palisimétrica*. Las dos mitades ga y ga tienen los mismos caracteres con la misma frecuencia. Otros ejemplos son rotor, aabccaba, xyzxy. La palabra abbaab NO es *palisimétrica*.

En este ejercicio, una palabra se representa mediante la clase `SecuenciaCaracteres`. La descripción aparece en la tabla incluida más abajo y no es necesario implementar los métodos que aparecen en dicha tabla. Supondremos que los caracteres almacenados pertenecen al conjunto de letras minúsculas consecutivas del alfabeto inglés (abcdefghijklmnopqrstuvwxyz).

Tareas a realizar

a) Implemente los siguientes métodos en la clase:

- **(0.5 puntos)** `SubSecuencia` que devuelve una secuencia de caracteres compuesta por los caracteres comprendidos entre dos posiciones dadas `inicio` y `final`. Deberá considerar todas las situaciones posibles que pueden ocurrir con los valores de `inicio` y `final`.
- **(4.5 puntos)** `EsPalisimetrica` que comprueba si la secuencia representa una palabra *palisimétrica*. Explique brevemente cómo ha resuelto la tarea.

Debe considerar que la secuencia puede contener muchos caracteres, por lo que el método debe ser lo más **EFICIENTE** posible, en términos de la velocidad de ejecución. Considere que no hay restricciones de memoria.

Para ambos métodos indique la cabecera completa, así como sus parámetros y datos locales. Puede añadir todos los métodos auxiliares que considere. Si lo hace, justifique su necesidad y ámbito (público, privado).

b) Importante: Construya una palabra utilizando la concatenación de su nombre y primer apellido (todo en minúsculas, sin acentos ni espacios) y explique con claridad el funcionamiento de su método `EsPalisimetrica` sobre esa palabra. Por ejemplo, si su nombre es "Simón Fernández Alonso", deberá usar `simonfernandez`. No se le pide que construya la función `main` sino que explique el funcionamiento de su método usando dicha palabra.

SecuenciaCaracteres
<i>Datos miembros privados:</i>
<code>static const int TAMANIO = 5e6</code> <code>char v[TAMANIO]</code> <code>int utilizados</code>
<i>Métodos públicos disponibles (no debe implementarlos):</i>
<code>SecuenciaCaracteres()</code> <code>int Utilizados()</code> <code>int Capacidad()</code> <code>void Aniade(int nuevo)</code> <code>char Elemento(int indice)</code> <code>int PrimeraOcurrencia(char a_buscar)</code>

◁ Ejercicio 2 ▷ Tablero y sus Propiedades

[5 puntos]

Un tablero T es un cuadrado de dimensiones $N \times N$ donde se ubican valores enteros (positivos y negativos). Se dice que T es un tablero *heterogéneo* si sus columnas tienen sumas **diferentes**. Además, si un tablero es *heterogéneo* entonces puede cumplir una propiedad adicional: ser *heterogéneo completo*. Para ello, se necesita que con los valores ordenados de las sumas de las columnas se pueda formar una secuencia creciente completa. Es decir, dada la secuencia $S = \{s_1, s_2, \dots, s_k\}$, decimos que es creciente y completa si $s_{i+1} = s_i + 1$.

A continuación se muestran dos ejemplos. A la izquierda, un Tablero *heterogéneo*. Podemos ver que las columnas dan lugar a sumas diferentes. A la derecha, un Tablero *heterogéneo completo*. Todas las sumas son diferentes (es *heterogéneo*) y además, al ordenarlas obtenemos una secuencia creciente completa de valores $\{13, 14, 15\}$.

$$\begin{bmatrix} 9 & 8 & 7 \\ 2 & 1 & 6 \\ 3 & 4 & 5 \\ \downarrow & \downarrow & \downarrow \\ 14 & 13 & 18 \end{bmatrix}$$

$$\begin{bmatrix} 9 & 8 & 7 \\ 2 & 1 & 6 \\ 3 & 4 & 2 \\ \downarrow & \downarrow & \downarrow \\ 14 & 13 & 15 \end{bmatrix}$$

Dado un tablero T , se quiere implementar un programa para saber si es *heterogéneo* y *heterogéneo completo*. Para ello, dispone de dos clases: Tablero y SecuenciaEnterosLong que puede utilizar, sin necesidad de implementar los métodos que aparecen en los diagramas de abajo.

Tablero
<i>Datos miembros privados:</i>
static const int MAX = 1e3 long t[MAX][MAX] int dimension
<i>Métodos públicos disponibles:</i> (no debe implementarlos):
Tablero(int dimension) int Dimension() long Elemento(int fil, int col) void Modifica(int fil, int col, long valor)

SecuenciaEnterosLong
<i>Datos miembros privados:</i>
static const int TAMANIO = 1e3 long v[TAMANIO] int utiles
<i>Métodos públicos disponibles</i> (no debe implementarlos):
SecuenciaEnterosLong() int Utilizados() int Capacidad() void Aniade(long nuevo) long Elemento(int indice) int PrimeraOcurrencia(long valor) void OrdenaCreciente()

Tareas a realizar

a) (0.5 puntos) Implemente un método para obtener la diagonal inversa de un tablero como un objeto de la clase SecuenciaEnterosLong. Con cualquiera de los tableros de la figura de arriba, la secuencia resultante sería $\{7, 1, 3\}$

b) (4.5 puntos) Implemente los siguientes métodos:

- Heterogeneo que comprueba si el tablero cumple la propiedad de ser *heterogéneo*
- HeterogeneoCompleto que comprueba si el tablero cumple la propiedad de ser *heterogéneo completo*.

En ambos métodos debe considerar que los métodos deben ser lo más **EFICIENTES** posible, en términos de la velocidad de ejecución. Considere que no hay restricciones de memoria.

Para ambos métodos indique la cabecera completa, así como los datos locales. Explique brevemente cómo ha resuelto las tareas pedidas.

Puede añadir todos los métodos que considere oportunos. Deberá implementarlos completamente, indicando también el ámbito (público, privado) de los mismos. **SIGUE POR LA SIGUIENTE PÁGINA**



c) Importante: Construya un tablero de 3×3 utilizando los dígitos de su DNI y reemplazando la letra por el dígito 3 de la siguiente manera: si su DNI es 12345678K, entonces el tablero será:

1 2 3
4 5 6
7 8 3

Escriba en el examen el tablero asociado a su DNI y explique con claridad cómo funciona su método `HeterogeneoCompleto`.