

## PROGRAMACIÓN I - GRUPO A

### PRÁCTICA Nº2

17 de octubre 2023

---

#### INSTRUCCIONES

---

1. Se debe acceder a la **SALA DE GRUPO** del campus virtual de vuestro equipo de trabajo.
2. Durante el desarrollo de la sesión práctica se debe seguir el procedimiento de trabajo en la sala asignada para realizar las actividades evaluables. Recordad que es **imprescindible GRABAR toda la sesión y COMPARTIR la pantalla completa**.
3. La entrega de la práctica sólo se admitirá a través de la **actividad** disponible en el campus virtual de la asignatura de **Prácticas de Programación I antes de la hora de finalización de la sesión de prácticas**.
4. Se debe entregar **un único fichero en formato .txt o .cpp, sin comprimir**, con el nombre de **P2GA**.
5. El fichero entregado debe **incluir el nombre de los integrantes del equipo**.
6. Aunque las prácticas se realizan en grupos de dos integrantes, para su evaluación, **ambos deberán hacer la entrega a través de su campus virtual**. En otro caso, la práctica quedará **sin evaluar** y supondrá un **0 en su calificación**.
7. El incumplimiento de alguna de las instrucciones sobre la realización/entrega de la Práctica supondrá su **descalificación**.

## A. TEST [30% puntuación] – 20 min

---

Realizar de **forma individual** desde el campus virtual de la asignatura accediendo al apartado **Enunciado PII/Test Práctica N2-PI Grupo A (30%)**.

- **No se permite ningún tipo de material de apoyo.**
- El test consta de **10 preguntas** que se mostrarán de **una en una en orden aleatorio**.
- Puntuación por pregunta:
  - Correcta +1 punto
  - No contestado 0 puntos
  - Incorrecta -0.25 puntos.
- **Tiempo 20 minutos máximo.**
- Una vez iniciado el test, se debe terminar de una sola vez.
- Si hay algún problema técnico, comunicarlo al profesor/a.

## B. EJERCICIO PRÁCTICO [70% puntuación] – 150 min

---

Escriba un programa en C++11, **P2GA**, utilizando los tipos de datos, las operaciones de entrada y salida y las estructuras de control vistas hasta ahora, que muestre por pantalla un menú con las siguientes opciones:

1. **Conversor de números arábigos a números romanos.**
2. **Determinar las cifras de un número.**
3. **Determinar número de vocales.**
4. **Salir programa.**

Siempre que se seleccione cualquiera de las primeras opciones el programa deberá realizar dicha operación y volver a presentar de nuevo el menú anterior para que el usuario pueda repetir el proceso seleccionando otra opción. Si el usuario selecciona la **última opción**, el programa terminará mostrando el mensaje: **“Gracias y hasta pronto”**.

Si la opción indicada por el usuario no es ninguna de las anteriores, se mostrará el mensaje **“La opción no es válida. Por favor, vuelva a seleccionar otra opción”**.

**(1,5 puntos)**

**Importante:** Las diferentes opciones deben ser resueltas de forma algorítmica, es decir, la solución propuesta tiene que ser general y no particular para unos determinados datos.

La descripción de las opciones del menú es la siguiente:

### 1. Conversor de números arábigos a números romanos (3 puntos)

Esta opción consistirá en **solicitar un número entero positivo al usuario (comprendido entre 1 y 99)** para **determinar su representación en números romanos y mostrarla por pantalla.**

Por ejemplo, si el número tecleado por el usuario es 84, se mostrará por pantalla:

***84 en romano se escribe: LXXXIV***

**Información adicional:** La numeración romana se basa en el empleo de siete letras del alfabeto latino a las que corresponde un valor numérico fijo. Para el ejercicio es suficiente utilizar cinco de estas letras:

I=1	V=5	X=10	L=50	C=100
-----	-----	------	------	-------

Las reglas para escribir números romanos son:

1. Se escriben siempre en mayúsculas.
2. Las letras M, C, X, I no se pueden repetir más de tres veces consecutivas. Por ejemplo, 33 se escribe: XXXIII y 44 debe escribirse: XLIV y no XXXXIIII.
3. Las letras V y L se pueden colocar a la derecha de otro signo para sumar su valor, pero sólo una vez, no se pueden repetir: 15 debe escribirse XV, y no VVV.
4. Cuando un signo va seguido de otro valor igual o inferior, se suman sus valores: VI=6, XV=15, XXVII=27.
5. Cuando los signos I, X, C van seguidos de otro de mayor valor, se restan del valor de este: IX=9, XL=40.
6. Los signos V y L no se utilizan nunca con valor de resta colocándolas a la izquierda de un valor superior: 45 debe escribirse XLV, y no VL.

## 2. Determinar las cifras de un número (2,5 puntos)

Esta opción consistirá en solicitar un **número entero positivo al usuario** para determinar sus cifras y guardar en una variable de tipo **struct** los campos que seguidamente se describen y cuyo contenido se debe de mostrar por pantalla.

La variable de tipo **struct** debe contener los siguientes campos:

- **cifraMenor**: campo que almacenará la menor de las **cifras** que tiene el número solicitado.
- **cifraMayor**: campo que almacenará la mayor de las **cifras** que tiene el número solicitado.
- **totalCifras**: campo que almacenará el total de cifras que tiene el número solicitado.

El proceso debe realizar las siguientes tareas:

- Solicitar un **único número entero positivo**.
- Determinar, a través de las operaciones de **división (/)** y **resto de división entera (%)**, las cifras que forman el número y procesar cada una de ellas para obtener la información de la **struct** descrita anteriormente:
  - Determinando si se debe actualizar alguno de los campos **cifraMenor** o **cifraMayor**, según corresponda.
  - Incrementando el campo **totalCifras**.
- Mostrar por pantalla el número solicitado al usuario y la información obtenida sobre sus cifras (contenido de los campos **cifraMenor**, **cifraMayor** y **totalCifras** de la **struct**).

## 3. Determinar el número de vocales (2 puntos)

Esta opción consistirá en solicitar tantas **vocales al usuario** como quiera teclear por teclado hasta que introduzca un carácter que no sea una vocal. Cada vocal servirá para incrementar el total de cada vocal que se incluirá en una variable de tipo **struct** y cuyo contenido se debe de mostrar por pantalla.

La variable de tipo **struct** debe contener los siguientes campos:

- **totalA**: campo que almacenará el total de vocales A/a que se han tecleado.
- **totalE**: campo que almacenará el total de vocales E/e que se han tecleado.
- **totalI**: campo que almacenará el total de vocales I/i que se han tecleado.
- **totalO**: campo que almacenará el total de vocales O/o que se han tecleado.
- **totalU**: campo que almacenará el total de vocales U/u que se han tecleado.

El proceso debe realizar las siguientes tareas:

- Solicitar **vocales** hasta que se introduzca una letra que no sea vocal.
- Por cada vocal se incrementará el correspondiente campo de la variable **struct** donde se lleva la cuenta para dicha vocal.
- Cuando se introduzca un carácter diferente a una vocal, se mostrará por pantalla el contenido de la variable **struct** (contenido de los campos que son los contadores para cada vocal).

## RUBRICA DE CALIFICACIÓN

- El 30% de la nota de la práctica 2, simulacro de examen, vendrá dado por el test (10 preguntas, 20 minutos). Esta puntuación será individual.
- El 70% restante será de la práctica entregada (misma nota para cada integrante de manera general), siguiendo la siguiente rúbrica para cada apartado:

Se debate en equipo sobre la estrategia a seguir, haciendo un <b>boceto de la estructura</b> que tendrá el programa (a papel, en la siguiente hoja).	1 punto
El programa muestra las <b>diferentes opciones del menú</b> , permitiendo al usuario seleccionar una opción, y repitiéndose hasta que se pulse salir.	1,5 puntos
Opción 1: <b>Conversor de números arábigos</b>	
Petición y validación del número	0,75 puntos
Conversión a números romanos.	1,5 puntos
Visualización.	0,75 puntos
Opción 2: <b>Determinar las cifras de un número</b>	
Determinación de las cifras del número	1 punto
Definir <b>struct</b> . Procesar la cifra y almacenarla en la <b>struct</b>	1 punto
Visualización del número y del contenido de la <b>struct</b>	0,5 puntos
Opción 3: <b>Determinar el número de vocales</b>	
Solicitar letra y controlar final de petición	0,5 puntos
Definir <b>struct</b> . Contabilizar vocal en la <b>struct</b>	1 punto
Visualización el contenido de la <b>struct</b>	0,5 puntos

Además, cada apartado de la parte práctica, se evaluará considerando lo siguiente:

	% máx. (*)
El elemento evaluable no compila o no se asemeja a lo que se pide	0%
El elemento evaluable no se aproxima suficientemente a lo pedido	40%
El elemento evaluable se aproxima suficientemente a lo pedido	60%
El elemento evaluable funciona correctamente y las estrategias y elementos de código elegidos son adecuados.	100%

**(\*) El porcentaje (% max.) representa el valor máximo sobre el que se evalúa el elemento indicado.**