

PROGRAMACIÓN I- GRUPO B

PRÁCTICA Nº5

21 de diciembre 2023 18:30-21:30

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 **P4Ejercicio**.
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**.

Al final del enunciado de la práctica tenéis la puntuación de los diferentes hitos que debéis cumplir durante la sesión. Se recomienda que realicéis de forma previa al desarrollo una lectura comprensiva del enunciado, así como un debate en cada equipo con el que podáis planificar la estrategia a seguir durante la práctica.

Dicha estrategia deberá quedar plasmada en un flujograma que deberá ser entregado en papel al finalizar la práctica, indicando en el los nombres de los integrantes del equipo.

Recordad que podéis/debéis utilizar el chuleterio durante la sesión, y que debéis tener en cuenta la rúbrica general de evaluación. Por ser una práctica podéis consultarla durante la sesión sin problema.

EJERCICIO PRÁCTICO [70% puntuación] – 160 minutos

Apartado 1: [1 punto]

Definición del tipo de datos que creas oportuno para poder representar en la memoria del computador una carta de la baraja española. Teniendo en cuenta que la baraja española está formada por cuatro palos (oros, espadas, copas y bastos) y cada palo por un máximo de 12 números (1,2,3,...11 y 12).

Apartado 2: [1 punto]

Definición del tipo de datos que creas oportuno para poder representar en la memoria del computador un jugador de cartas. Estando compuesto por una variable que sea capaz de almacenar un nombre y una variable del tipo contenedor que consideres oportuno de etiqueta **Mano** que sea capaz de almacenar un número indeterminado de cartas.

Apartado 3: [3 puntos = (0,5 declaración + 2,5 definición)]

Implementar una función que genere una baraja española completa si recibe como parámetro de entrada un valor true y una baraja española reducida si recibe un valor false.

La versión completa estará formada por 48 naipes (4 palos de 12 cartas cada uno, numeradas del 1 al 12) y la reducida por 40 (4 palos de 10 cartas cada una, numeradas por 1, 2, 3, 4, 5, 6, 7, 10, 11 y 12)

Apartado 4: [3 puntos = (0,5 declaración + 2,5 definición)]

Implementar una función que, a partir de un jugador, un contenedor de cartas de etiqueta **Mazo** y una variable entera de etiqueta **N** realice un reparto aleatorio de N cartas al Jugador. De forma que las cartas del mazo seleccionadas aleatoriamente queden registradas en el campo **Mano** del jugador y sean eliminadas del **Mazo**.

En el caso que el número de cartas a repartir (**N**) sea superior al número de cartas disponible en el **Mazo** la función devolverá un valor **false** y en caso de poder haber realizado el reparto devolverá un valor **true**.

Apartado 5: [2 puntos]

Apartado 5.1 [1,2 puntos]: en la función principal del programa declarar un array de 4 jugadores, inicializando sus nombres a “Player1”, “Player2”, “Player3” y “Player4” y utilizando la función del apartado 4 realizar el reparto aleatorio de 10 cartas de un mazo completo a cada jugador.

Apartado 5.2 [0,8 puntos]: utilizando la información almacenada en el array del apartado 5.1 crear un vector de punteros inteligentes que apunten a zonas de memoria del sistema que almacenen la misma información.

Para la calificación de este apartado no se tendrá en cuenta que la definición de la función del apartado 4 sea correcta. En caso de no lograr su funcionamiento se podrá presentar una propuesta de solución que haga uso del prototipo planteado, siendo necesario incluir un comentario con una breve descripción del mismo.

RÚBRICA DE CALIFICACIÓN

- El 30% de la nota de la práctica, se obtendrá sobre trabajo realizado durante la sesión (resolución de preguntas, dominio de la materia, etc.). Esta nota será individual, y se considerarán para su valoración aspectos como:
 - **Defensa individual** a cuestiones que plantee el profesor durante la sesión a cada integrante del equipo.
 - **Trabajo/debate en equipo** sobre el desarrollo a realizar en la práctica. Quedando representado en un flujograma en el que se visualice el concepto de las funciones indicadas en los apartados 3 y 4. **Deberá desarrollarse al inicio de la práctica y se entregará en papel a la finalización de la misma.**
- El 70% restante será sobre la puntuación del trabajo entregado antes de la finalización de cada práctica (misma nota para cada integrante del equipo), siguiendo la siguiente rúbrica general para cada apartado puntuable:

	%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 elegido son adecuados.	100%

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