

PRACTICA 1.

El alumno deberá desarrollar un programa que permita hacer operaciones con matrices. El programa deberá permitir:

1. Introducir por teclado una matriz del número de filas y columnas que el usuario desee. El usuario podrá introducir múltiples matrices. Estas matrices se irán almacenando en memoria RAM, y cada una de ellas tendrá un identificador asociado (por ejemplo, un número que vaya incrementando consecutivamente). (1,0 puntos)
2. Mostrar en la consola una matriz de las que están almacenadas. Para ello el usuario introducirá el identificador asociado con dicha matriz. (0,5 puntos)
3. Borrar una de las matrices almacenadas. El usuario introducirá el identificador asociado con la matriz que desea borrar. La memoria asignada a dicha matriz pasará a estar disponible para otra matriz. (0,5 puntos)
4. Sumar dos de las matrices que se hayan introducido previamente. Deberá comprobarse si ambas matrices se pueden sumar (tienen el mismo número de filas y de columnas). Al realizar la operación, además de mostrar por pantalla la matriz resultado, esta matriz quedará almacenada también en el entorno como una de las matrices disponibles para operar. (0,5 puntos)
5. Restar dos de las matrices que se hayan introducido previamente. Deberá comprobarse si ambas matrices se pueden restar (tienen el mismo número de filas y de columnas). Al realizar la operación, además de mostrar por pantalla la matriz resultado, esta matriz quedará almacenada también en el entorno como una de las matrices disponibles para operar. (0,5 puntos)
6. Multiplicar dos de las matrices que se hayan introducido previamente. Deberá comprobarse si ambas matrices se pueden multiplicar (el número de columnas de la primera coincide con el número de filas de la segunda). Al realizar la operación, además de mostrar por pantalla la matriz resultado, esta matriz quedará almacenada también en el entorno como una de las matrices disponibles para operar. (1,0 puntos)
7. El usuario elegirá una de las matrices disponibles, que necesariamente deberá ser cuadrada. Esa matriz se considerará como un sistema de N ecuaciones lineales con N incógnitas, y deberá resolverse por el método de Gauss-Jordan, consistente en hacer ceros en el triángulo inferior de la matriz. (1,5 puntos)
8. Guardar en un archivo en el disco duro, cuyo nombre será especificado por el usuario, todas las matrices disponibles en el programa en un momento dado. (1,5 puntos)
9. Cargar de un archivo en el disco duro, cuyo nombre será especificado por el usuario, un conjunto de matrices que hayan sido guardadas previamente. (1,5 puntos)

Si el alumno usa memoria dinámica para el almacenamiento de las matrices, de tal modo que no haya ningún tipo de límite en el número de matrices que se puedan almacenar, o en el tamaño de estas, el alumno obtendrá hasta 1,5 puntos extra.

Si el alumno lleva a cabo un diseño modular adecuado de la práctica, a través del uso de funciones y al menos dos módulos, el alumno podrá obtener hasta 1,5 puntos extra. Uno de los dos módulos en cuestión debería encargarse de realizar las operaciones con matrices, sin leer ni mostrar nada en la consola, y el otro módulo debería de actuar como de interfaz de usuario, encargándose de leer y mostrar datos en la consola.

Normas de entrega y valoración sobre la nota final

La práctica debe ser entregada tanto en formato electrónico como impresa en papel antes del 11 de enero (inclusive). Los alumnos que realicen el examen final de programación en modalidad online podrán presentar sólo una versión en formato electrónica de la práctica, no siendo necesaria la versión impresa. El alumno deberá realizar, cuando el profesor lo requiera, una defensa oral de la práctica y la calificación de la práctica será la obtenida en dicha defensa oral. Si el alumno no realiza dicha defensa oral tendrá 0 en la práctica, y por consiguiente no superará la asignatura. **Si durante la defensa de dicha práctica se demuestra que ha habido plagio en su realización, siguiendo la normativa actual de plagios de la Universidad, el alumno suspenderá la asignatura de programación en convocatoria ordinaria y extraordinaria, y tendrá que volver a cursar la asignatura el próximo curso académico.**

Dentro de los aspectos que se valorarán a la hora de decidir la nota de la práctica cabe destacar la utilización de funciones, uso de comentarios para acompañar al código, uso de nombres de variables significativos, presentación en pantalla, etc. La escritura de código enrevesado, difícil de leer, o el no usar funciones, etcétera bajará puntuación en la práctica.

No es una buena estrategia que comiences a trabajar en la práctica hasta que no tengas bien dominados los conceptos teóricos que involucra la funcionalidad de la práctica, y hayas hecho ejercicios sencillos para practicar dichos conceptos. Intentar abordar directamente la práctica, aunque pudiera parecer un camino más corto, es un camino mucho más largo y arduo.