

 <b>UNLaM</b> Dto. Ingeniería e Investigaciones Tecnológicas	Tópicos de programación 03635 Parcial Turno Tarde Martes 14-18 TT 19/06/2023
Apellido y Nombre:	
DNI:	
Calificación:	

## Parcial Tópicos de programación

Comisión: 4-3600

**ATENCIÓN:** para que el examen pueda ser corregido y archivado debe ser entregado obligatoriamente antes de la hora límite de las 18:00 teniendo en cuenta un horario de inicio 15:00 y con el siguiente formato: DNI\_APELLIDO\_NOMBRE.zip, ejemplo 41127133\_PEREZ\_MARIA\_PIA\_P.zip. Observe que es un .zip, no es .rar ni .7z respete la forma de entrega. La P como sufijo indica que es el PARCIAL. Adjunte el enunciado y elimine las carpetas *bin* y *obj*, está compartiendo programas y por obvias razones los servidores eliminarán comprimidos que contengan binarios o ejecutables.

Se entrega junto con este documento un proyecto desarrollado en codeblocks con los prototipos de una serie de funciones en el archivo examen.h. El archivo examen.c contiene los cuerpos de las funciones, pero no está desarrollada la implementación. Implemente las funciones sin modificar su firma.

Para su examen las funciones a implementar son:

- **mstrstr:** debe realizar la misma función que **strstr**, siendo esta una función de biblioteca ANSI C verifique en el pdf adjunto la funcionalidad y **replíquela fielmente**.
- **trasponerMatrizCuadrada:** traspone insitu una matriz cuadrada de enteros.
- **insertarOrdenado:** implemente una función que inserte ordenadamente y sea **GENÉRICA**. Se podrán insertar elementos de cualquier tipo que se mantendrán ordenados durante la inserción hasta colmar la capacidad del vector. Deberá operar de la siguiente manera:
  - Suponiendo orden ascendente, si el elemento a insertar es menor que el mayor elemento del vector deberá ingresar al vector y se retornará un 0/falso indicando que no ha sido una inserción perfecta, tuvimos pérdida de datos.
  - Suponiendo orden ascendente, si el elemento a insertar es mayor que el mayor elemento del vector el elemento a insertar no ingresará y se retornara 0/false indicando que no fue inserción.
  - En cualquier otro caso el elemento ingresa y se retorna 1/true, inserción limpia.

En el vector que se muestra a continuación suponiendo que tiene un máximo de 5 lugares disponibles el número 13 ingresaría y perderíamos el 16 y el número 17 no ingresaría. En ambos casos el valor de retorno será 0.

3	6	9	12	16
---	---	---	----	----

Recuerde que la implementación debe ser genérica, si el usuario necesita orden inverso las condiciones también respetaran comportamiento inverso. Si le sirve, piénselo como un ranking que debe conservar o los **n** mayores o lo **n** menores según necesidad del usuario.

La descripción de cada uno de los argumentos se encuentra en los fuentes entregados. La promoción exige el correcto desarrollo de todos lo puntos.