Instituto Tecnológico de Costa Rica **Escuela de Matemática**
Cátedra: Álgebra Lineal para Computación **I Semestre, 2017**

TAREA PROGRAMADA 2

Valor: 10 %

INDICACIONES GENERALES

- Esta tarea consiste en desarrollar tres programas según las especificaciones dadas.
- La tarea deberá ser resuelta en grupos de 4 o 5 estudiantes.
- Se podrá utilizar cualquier lenguaje de programación, tanto para el desarrollo de los algoritmos de programación, como para la visualización de la información (interfaz gráfica).
- La fecha límite para la entrega es el DOMINGO 4 de JUNIO, antes de las 11:50 pm. Después de la fecha y hora de cierre será imposible entregar la tarea.
- Se deberá subir una carpeta con el proyecto (que debe incluir el **archivo ejecutable de cada programa** y su respectivo **manual de usuario** en formato pdf) al Folder **Tarea Programada 2** que se encuentra en la sección **Tareas** de la Comunidad **Cátedra de Álgebra Lineal** del TecDigital). El nombre de la carpeta deberá ser **Grupo(número del grupo respectivo)Tarea 1**, por ejemplo **Grupo60Tarea1**.
- En caso de comprobar fraude en la solución de la tarea, se aplicarán las normativas internas vigentes del ITCR.

OBJETIVOS A EVALUAR

- Emplear el concepto de base y dimensión de un espacio vectorial.
- Comprender conceptos del álgebra vectorial.
- Emplear transformaciones lineales para resolver problemas en diferentes contextos.
- Fomentar la capacidad de análisis y de razonamiento deductivo.
- Promover el desarrollo de habilidades para la resolución de problemas utilizando herramientas de programación.

PARTE I. BASES Y DIMENSIÓN DEL ESPACIO SOLUCIÓN DE UN SISTEMA HOMOGÉNEO DE ECUACIONES LINEALES

Se debe crear un programa que sea capaz de dar la dimensión y una base del espacio de soluciones de un sistema homogéneo de ecuaciones lineales.

Para comenzar, se debe dar al usuario la opción de definir el tamaño del sistema con el que desea trabajar. Debe tenerse presente que el número de ecuaciones no necesariamente será igual al número de variables. Tanto el número de ecuaciones como el de variables deberán ser enteros mayores que 1 y menores que 6. Se recomienda que, al solicitar al usuario seleccionar tanto el número de ecuaciones (filas), como el número de variables (columnas), se despliegue un menú con las opciones: 2, 3, 4, 5; para que el usuario únicamente tenga que marcar la opción de su elección. Si el programa no presenta un menú de este tipo, será preciso hacer las validaciones respectivas para asegurar que los valores ingresados por el usuario sean adecuados (si no lo son, debe advertirse sobre el error, con la especificación de cuáles son los valores permitidos).

Posteriormente se deberá solicitar al usuario que introduzca cada uno de los coeficientes del sistema, (entradas fraccionarias con numeradores y denominadores de hasta 4 dígitos cada uno). Se recomienda crear una interfaz que facilite al usuario la introducción de cada uno de los coeficientes.

$$\left(\begin{array}{ccccccc} \frac{\square}{\square}X_1 & + & \frac{\square}{\square}X_2 & + & \frac{\square}{\square}X_3 & = & 0 \\ \frac{\square}{\square}X_1 & + & \frac{\square}{\square}X_2 & + & \frac{\square}{\square}X_3 & = & 0 \\ \frac{\square}{\square}X_1 & + & \frac{\square}{\square}X_2 & + & \frac{\square}{\square}X_3 & = & 0 \end{array}\right)$$

El programa debe validar que cada uno de los datos introducidos en las casillas (tanto en las de numeradores, como en las de denominadores) sea un número entero (positivo o negativo). Se recomienda inicializar los denominadores en 1 (para que, en caso de que alguna de las entradas sea entera, el usuario no tenga necesariamente que escribir un 1 en el denominador correspondiente), con la posibilidad de modificar este denominador (sobreescribir el denominador), en caso de que la entrada correspondiente sea fraccionaria no entera.

$$\left(\begin{array}{ccccccc} \frac{\square}{1}X_1 & + & \frac{\square}{1}X_2 & + & \frac{\square}{1}X_3 & = & 0 \\ \frac{\square}{1}X_1 & + & \frac{\square}{1}X_2 & + & \frac{\square}{1}X_3 & = & 0 \\ \frac{\square}{1}X_1 & + & \frac{\square}{1}X_2 & + & \frac{\square}{1}X_3 & = & 0 \end{array}\right)$$

BASES Y DIMENSIÓN DEL ESPACIO SOLUCIÓN DE UN SISTEMA HOMOGÉNEO DE ECUACIONES LINEALES

Una vez que se han introducido todos los coeficientes del sistema, deberá mostrarse el botón: **Resolver**. Si el sistema tiene solución única, el programa deberá mostrar dicha solución, usando formato vectorial con coordenadas fraccionarias, tal como se muestra a continuación:

$$\textbf{El sistema tiene solución única. Su solución es:} \begin{pmatrix} X_1 \\ \\ X_2 \\ \\ X_3 \end{pmatrix} = \begin{pmatrix} \frac{3}{5} \\ \\ \frac{-2}{1} \\ \\ \frac{1}{2} \end{pmatrix}$$

En caso de que el sistema tenga infinitas soluciones, el programa deberá indicar cuál es la dimensión de su espacio solución y mostrar una base para dicho espacio, tal como se ve en el siguiente ejemplo:

El sistema tiene infinitas soluciones.

La dimensión del espacio solución del sistema es 2.

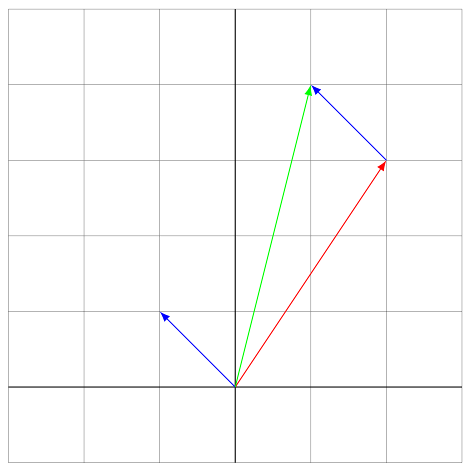
$$\textbf{Una base para el espacio solución del sistema es:} \left\{ \begin{pmatrix} \frac{1}{2} \\ \\ \frac{1}{1} \\ \\ \frac{0}{1} \end{pmatrix}, \begin{pmatrix} \frac{1}{3} \\ \\ \frac{0}{1} \\ \\ \frac{1}{1} \end{pmatrix} \right\}$$

Después del cálculo de la solución del sistema de ecuaciones respectivo, el programa deberá permitir al usuario ingresar un nuevo sistema de ecuaciones (sin tener que reiniciar el programa) o modificar los coeficientes del sistema ingresado originalmente. Esta última es una opción muy útil cuando el usuario se da cuenta de que cometió un error al ingresar los coeficientes respectivos, pues le da la posibilidad de hacer los cambios correspondientes sin tener que ingresar todos los datos de nuevo.

PARTE II. VISUALIZACIÓN DE VECTORES Y TRANSFORMACIONES LINEALES EN $\mathbb{R}^2$

Un vector $v = (a, b) \in \mathbb{R}^2$ puede visualizarse como un segmento de recta dirigido, con su extremo inicial en el origen $(0, 0)$ de un sistema de coordenadas cartesianas y con su extremo final en el punto (a, b) de dicho sistema (suele emplearse una flecha para designar al extremo final del vector). De forma más general, dicho vector puede representarse como un segmento de recta dirigido, con su extremo inicial en cualquier punto (r, s) del sistema de coordenadas cartesianas y con su extremo final en el punto $(r + a, s + b)$ de dicho sistema.

En la siguiente gráfica se muestran los vectores $(2, 3)$ **(en rojo)** y $(-1, 1)$ **(en azul)**.



Observe que al desplazar el vector $(-1, 1)$, de modo que su extremo inicial coincida con el extremo final del vector $(2, 3)$, se obtiene una representación gráfica de la suma de los vectores $(2, 3)$ y (-1.1). En este caso, $(2, 3) + (-1, 1) = (1, 4)$ **(en verde)**.

También es posible representar de un modo similar la multiplicación de un escalar por un vector y la diferencia de vectores.

Se debe crear un programa que evalúe una transformación lineal $T : \mathbb{R}^2 \to \mathbb{R}^2$ en una expresión vectorial del tipo $\alpha u + \beta v$ donde $\alpha, \beta \in \mathbb{R}$ y $u, v \in \mathbb{R}^2$. El programa deberá mostrar una representación gráfica de esta transformación.

Primeramente deberá solicitarse al usuario introducir los vectores u y v de $\mathbb{R}^2$; los escalares α y β; y la matriz 2×2 que representa a la transformación lineal $T : \mathbb{R}^2 \to \mathbb{R}^2$. En todos los casos, las entradas serán números enteros de a lo sumo dos dígitos y será necesario que se hagan las validaciones respectivas. En caso de que una o más de las entradas ingresadas por el usuario sean inválidas, deberá mostrarse un mensaje que lo indique y permitirle reescribir correctamente sobre aquellas entradas inválidas, sin que tenga que volver a ingresar las que sí son correctas.

La matriz de la transformación deberá estar inicializada como la matriz identidad $\begin{pmatrix} \boxed{1} & \boxed{0} \\ \boxed{0} & \boxed{1} \end{pmatrix}$, pero con la posibilidad de que el usuario cambie los valores de sus entradas, si así lo desea.

VISUALIZACIÓN DE VECTORES Y TRANSFORMACIONES LINEALES EN $\mathbb{R}^2$

Deberá mostrarse un botón con la opción **"CALCULAR Y GRAFICAR"** para ejecutar el programa.

Para este programa se debe crear una interfaz de cuatro ventanas, cada una con su encabezado correspondiente, tal como se muestra en la siguiente imagen:

Vectores u **y** v	**Vectores** $T(u)$ **y** $T(v)$
Vectores αu **,** βv **y** $\alpha u + \beta v$	**Vectores** $T(\alpha u)$ **,** $T(\beta v)$ **y** $T(\alpha u + \beta v)$

Introduzca los valores correspondientes

$$\alpha = \square \qquad \beta = \square \qquad u = (\square, \square) \qquad v = (\square, \square) \qquad T = \begin{pmatrix} \boxed{1} & \boxed{0} \\ \boxed{0} & \boxed{1} \end{pmatrix}$$

.

CALCULAR Y GRAFICAR

.

- En la ventana superior izquierda se deben mostrar las gráficas de los vectores u, v.
- En la ventana inferior izquierda se visualizará la gráfica de los vectores αu , βv y $\alpha u + \beta v$.
- En la ventana superior derecha se presentarán las graficas de los vectores $T(u)$ y $T(v)$.
- En la ventana inferior derecha se mostrarán las gráficas de los vectores $T(\alpha u)$, $T(\beta v)$ y $T(\alpha u + \beta v)$.

En las ventanas ubicadas en la sección inferior deben emplearse efectos de desplazamiento (animaciones) que permitan al usuario observar que al sumar los vectores αu y βv, efectivamente se obtiene el vector $\alpha u + \beta v$; y que de igual manera, al sumar los vectores $T(\alpha u)$ y $T(\beta v)$, efectivamente se obtiene el vector $T(\alpha u + \beta v)$.

Es importante emplear la misma escala en todas las cuatro ventanas, para que el efecto de la transformación sobre los vectores sea más claro.

PARTE III. TRANSFORMACIONES APLICADAS A IMÁGENES DIGITALES

El manejo adecuado de imágenes digitales en diferentes contextos (videojuegos, fotografía, imágenes médicas, programas de diseño, etc.) exige la aplicación de diversas nociones del Álgebra Lineal. El uso de transformaciones lineales es frecuente en muchos de esos casos.

Uno de los primeros problemas que se presentan al intentar trabajar con imágenes digitales, se debe a la necesidad de obtener representaciones discretas de figuras continuas. Para comprender mejor la idea anterior, puede pensarse en la fotografía digital de un paisaje. En este caso, la imagen que se muestra en la fotografía está compuesta por una serie de puntos (pixeles) contenidos en un "conjunto bidimensional discreto", que intentan reproducir, de cierta forma, un objeto tridimensional y continuo.

El problema se agrava cuando se intenta aplicar alguna transformación geométrica sobre una figura digitalizada, en busca de obtener una nueva figura digitalizada. Simplificaremos el caso suponiendo que la primera figura está compuesta únicamente por puntos blancos y negros, por lo que su mapa de bits puede ser descrito únicamente mediante los pares ordenados de coordenadas de posición (x, y) de los pixeles negros. De esta forma, es posible aplicar transformaciones a cualquier figura, de manera tal que el color de cada pixel en la figura original sea el mismo que el de su correspondiente pixel de la figura transformada.

Si se aplica la transformación lineal dada por $L(x, y) = (x, 3y)$ a un "segmento de recta" con extremos ubicados en las posiciones (0, 500) y (0, 800), es fácil ver que esta transformación envía los extremos del "segmento" original a los puntos (0, 1500) y (0, 2400) en la nueva figura. De forma análoga, enviaría cualquier punto $(0, a)$ con $500 \leq a \leq 800$ de la figura original, a un punto $(0, 3a)$ en la figura transformada. Así, el punto (0, 603) del segmento original es transformado en el punto (0, 1809) de la figura transformada.

Intuitivamente esperaríamos que el segmento original sea transformado en un nuevo segmento de recta, sin embargo, es fácil ver que ni el pixel de coordenadas (0, 1810), ni el de coordenadas (0, 1811) en la nueva figura aparecerían coloreados de negro, a pesar de que ambos puntos están contenidos en el segmento de recta con extremos en los puntos (0, 1500) y (0, 2400). Lo anterior se debe a que ningún pixel de la primera figura tiene por imagen a uno de estos nuevos puntos. Este problema se presenta precisamente como consecuencia del intento por expresar un objeto continuo, mediante una representación discreta. Afortunadamente existen diversos algoritmos, como la interpolación bilineal, la interpolación bicúbica o la interpolación del vecino más cercano, para dar solución a este problema.

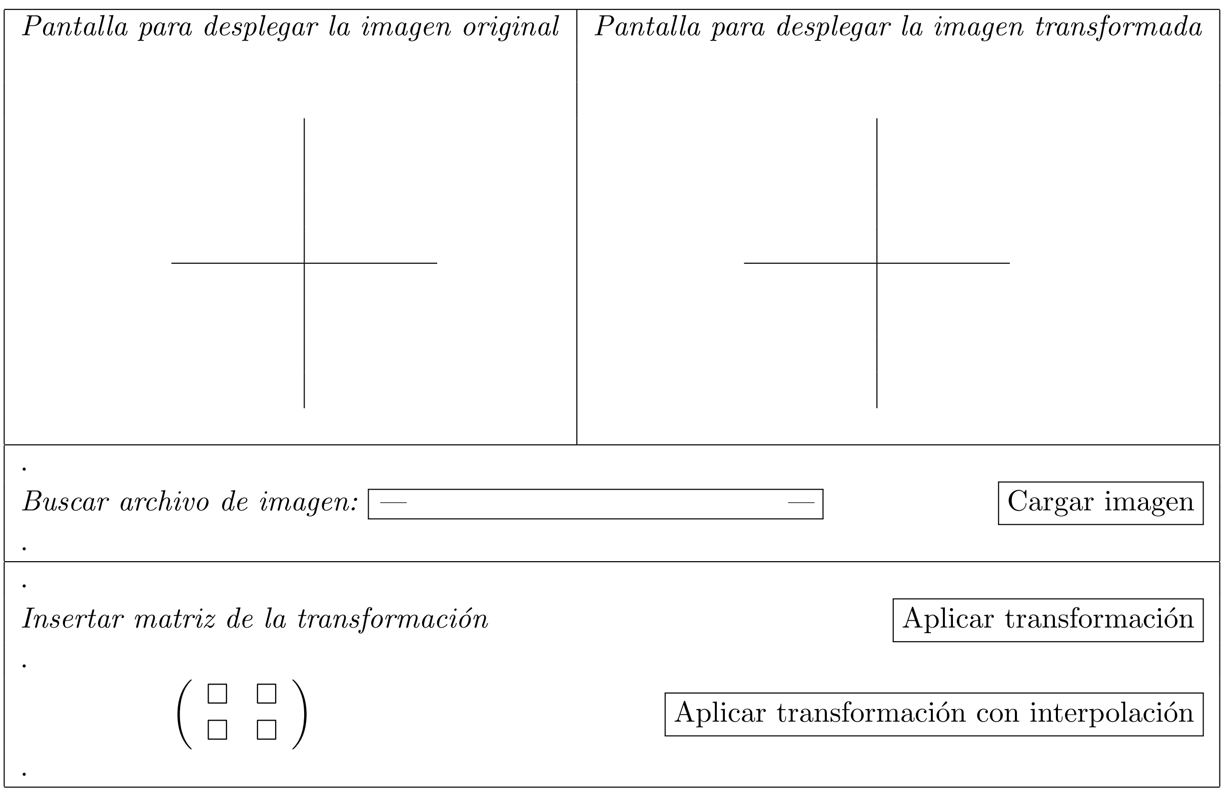
NOTA:
Se recomienda realizar una búsqueda de información sobre el problema propuesto. La lectura del artículo: **TRANSFORMACIONES GEOMÉTRICAS SOBRE IMÁGENES DIGITALES** (Barreto, s.f.) *y de algunos apartados de la tesis:* **REGISTRO DE IMÁGENES MEDIANTE TRANSFORMACIONES LINEALES POR TROZOS** (Arévalo, 2008) *puede ser de utilidad para alcanzar una mejor comprensión de todo lo mencionado anteriormente.*

TRANSFORMACIONES APLICADAS A IMÁGENES DIGITALES

Se debe crear un programa que sea capaz de aplicar transformaciones lineales a diversas imágenes digitales. El programa deberá recibir una imagen en formato PNG (en escala de grises) y desplegarla en pantalla, sobre un sistema de coordenadas rectangulares (coordenadas x-y). El usuario tendrá la opción de desplazar (horizontal y/o verticalmente) el origen del sistema de coordenadas cartesianas a cualquier punto sobre la imagen.

Adicionalmente deberá solicitarse al usuario que introduzca los coeficientes (las entradas serán números enteros) de una matriz 2×2. Esta matriz corresponde a la matriz de una transformación lineal. El programa debe aplicar dicha transformación a cada uno de los puntos de la figura (vistos como puntos con coordenadas x-y en el sistema de coordenadas definido por el usuario) y desplegar en pantalla la nueva imagen generada.

La interfaz debe permitir que el usuario vea tanto la imagen original y como la imagen obtenida después de efectuar la transformación. Se recomienda un formato de ventana doble, de tal forma que en la parte izquierda se muestre la imagen original introducida por el usuario y en la parte derecha se muestre la imagen obtenida después de aplicar la transformación correspondiente.



Es importante tener en cuenta la escala que se debe emplear al momento de desplegar la figura transformada. Idealmente, la nueva imagen deberá tener un tamaño que permita la visualización de todos los puntos obtenidos después de transformar la imagen original.

Se dará hasta un máximo de 10 puntos adicionales si el programa puede calcular (como opción adicional) la transformación, empleando alguno de los métodos de interpolación para mejorar la calidad de la imagen

CRITERIOS DE EVALUACIÓN

La tarea será evaluada de la siguiente manera:

1. **Manual de usuario** 10 puntos.

   Debe contener la siguiente información:

   - Número de grupo y nombre del profesor respectivo.
   - Nombre, apellidos y dirección de correo electrónico de cada uno de los integrantes del grupo.
   - Breve explicación sobre la forma en la que se deben ejecutar los programas y la manera de insertar los datos.
   - Cualquier otro que considere necesario.

2. **Programa UNO** 30 puntos
3. **Programa DOS** 30 puntos
4. **Programa TRES** 30 puntos

Total 100 puntos

En caso de tener alguna duda con lo que se solicita en esta tarea programada, o con la forma en que será evaluada, se le insta a plantearla en clases con su profesor de curso o en horas de consulta con cualquiera de los profesores de la cátedra.

Cátedra: ALC
I Semestre, 2017