Engrape aquí
No doble

Análisis Numérico I.
Examen de conocimientos previos. Variante 1.

Programación con arreglos, álgebra lineal, elementos de análisis.

Nombre: Calificación:

El examen dura 40 minutos. El resultado no afecta a ninguna calificación oficial. El propósito del examen es ayudar al estudiante comprender mejor qué tipos de ejercicios se resuelven en esta asignatura, identificar sus propias fortalezas y debilidades; luego, basándose en esta información, tomar una decisión consciente sobre la inscripción. Además, el examen ayudará al profesor identificar las fortalezas y debilidades de los estudiantes.

Problema 1. *10 %.*
Escribir una función en algún lenguaje de programación que resuelva sistemas de ecuaciones lineales con matrices unitriangulares inferiores bidiagonales. Por ejemplo, para $n = 5$ se trata del sistema

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ a_1 & 1 & 0 & 0 & 0 \\ 0 & a_2 & 1 & 0 & 0 \\ 0 & 0 & a_3 & 1 & 0 \\ 0 & 0 & 0 & a_4 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{bmatrix}.$$

No formar la matriz del sistema en la forma completa, trabajar solamente con los arreglos unidimensionales a, r y x.

- Entrada: arreglos a y r.
- Propiedades de la entrada: si denotamos la longitud de r por n, entonces la longitud de a es $n - 1$ (suponer que esta condición se cumple y no verificarla).
- Salida: arreglo x.

Calcular el número de operaciones de multiplicación que se realizan en el algoritmo, si el arreglo r tiene longitud n.

Problema 2. *10 %.*
En algún lenguaje de programación escribir una función que multiplique matrices cuadradas por vectores. Por ejemplo, para $n = 4$ se trata del producto

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \\ A_{2,1} & A_{2,2} & A_{2,3} & A_{2,4} \\ A_{3,1} & A_{3,2} & A_{3,3} & A_{3,4} \\ A_{4,1} & A_{4,2} & A_{4,3} & A_{4,4} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}.$$

- Entrada: arreglos A y b.
- Propiedades de la entrada: b es un arreglo unidimensional, y si denotamos la longitud de b por n, entonces A es una matriz de tamaño $n \times n$ (suponer que estas condiciones se cumplen y no verificarlas).
- Salida: arreglo p.

Calcular el número de operaciones de multiplicación que se realizan en el algoritmo, si el arreglo b tiene longitud n.

Problema 3. *10 %.*
Sea A una matriz real cuadrada de orden n. Escribir al menos 12 condiciones equivalentes a la condición que A es invertible. Notamos que en total se pueden encontrar más de 20 condiciones equivalentes a la invertibilidad de A.

Problema 4. *10 %.*
Sean a y b dos vectores del espacio $\mathbb{R}^n$. Se sabe que

$$\begin{aligned} &\|a\|_1 = 8, && \|a\|_2 = 5, && \|a\|_\infty = 3, \\ &\|b\|_1 = 11, && \|b\|_2 = 10, && \|b\|_\infty = 9. \end{aligned}$$

Usando esta información encontrar varias cotas superiores para $|a^\top b|$ y elegir la cota superior más precisa.

Engrape aquí
No doble

Análisis Numérico I.
Examen de conocimientos previos. Variante 2.

Programación con arreglos, álgebra lineal, elementos de análisis.

Nombre: Calificación:

El examen dura 40 minutos. El resultado no afecta a ninguna calificación oficial. El propósito del examen es ayudar al estudiante comprender mejor qué tipos de ejercicios se resuelven en esta asignatura, identificar sus propias fortalezas y debilidades; luego, basándose en esta información, tomar una decisión consciente sobre la inscripción. Además, el examen ayudará al profesor identificar las fortalezas y debilidades de los estudiantes.

Problema 1. *10 %.*
Escribir una función en algún lenguaje de programación que multiplique matrices bidiagonales triangulares superiores por vectores. Por ejemplo, para $n = 4$ se trata del producto

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} a_1 & 0 & 0 & 0 \\ b_1 & a_2 & 0 & 0 \\ 0 & b_2 & a_3 & 0 \\ 0 & 0 & b_3 & a_4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}.$$

No formar la matriz del sistema en la forma completa, trabajar solamente con los arreglos unidimensionales a, b, c y p.

- Entrada: arreglos de números a, b y c.
- Propiedades de la entrada: si denotamos la longitud de a por n, entonces la longitud de b es $n - 1$, y la longitud de c es n (suponer que estas condiciones se cumplen y no verificarlas).
- Salida: arreglo x.

Calcular el número de operaciones de multiplicación que se realizan en el algoritmo, si el arreglo a tiene longitud n.

Problema 2. *10 %.*
En algún lenguaje de programación escribir una función que resuelva sistemas de ecuaciones lineales con matrices unitriangulares superiores. Por ejemplo, para $n = 4$ se trata del sistema

$$\begin{bmatrix} 1 & A_{1,2} & A_{1,3} & A_{1,4} \\ 0 & 1 & A_{2,3} & A_{2,4} \\ 0 & 0 & 1 & A_{3,4} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix}.$$

En este problema la matriz A está dada en la forma completa, como un arreglo $n \times n$.

- Entrada: arreglos A y r.
- Propiedades de la entrada: si denotamos la longitud de r por n, entonces A es una matriz $n \times n$ unitriangular superior (suponer que estas condiciones se cumplen y no verificarlas).
- Salida: arreglo x.

Calcular el número de operaciones de multiplicación que se realizan en el algoritmo, si el arreglo r tiene longitud n.

Problema 3. *10 %.*
Sea A una matriz real cuadrada de orden n. Supongamos que las columnas de A forman una base ortonormal del espacio $\mathbb{R}^n$. Demostrar que los renglones de A también forman una base ortonormal del espacio $\mathbb{R}^n$.

Problema 4. *10 %.*
Sea A una matriz real cuadrada de orden 2. Definimos la función $q\colon \mathbb{R}^2 \to \mathbb{R}$ mediante la regla

$$q(x) = x^\top A\, x.$$

Calcular el gradiente de q. Escribir la respuesta en la forma matricial, es decir, en términos de la matriz A y del vector x.