

## IE-0217 Estructuras de Datos Abstractas y Algoritmos para Ingeniería

### Aplicaciones requeridas para C/C++ y manejo de reservorios

#### Tarea 02

#### Planteamiento de Problema

El objetivo de esta tarea es desarrollar un programa en C++ usando “clases” para resolver un sistema de ecuaciones de 4x4 mediante el método de GAUSS-JORDAN. según la siguiente descripción algorítmica.

1. Deben ordenar las ecuaciones de la siguiente forma:

$$a(00)*x + a(01)*y + a(02)*w + a(03)*z = a(04)$$

$$a(10)*x + a(11)*y + a(12)*w + a(13)*z = a(14)$$

$$a(20)*x + a(21)*y + a(22)*w + a(23)*z = a(24)$$

$$a(30)*x + a(31)*y + a(32)*w + a(33)*z = a(34)$$

2. Deben ordenar las ecuaciones de manera que los coeficientes  $a(00), a(11), a(22), a(33)$  sean diferentes de cero, de otra manera no podrán resolver las ecuaciones.
3. El algoritmo de solución es el siguiente:
  - a) Deben tomar la primera ecuación y definimos  $i$  como el índice que apunta a la ecuación en proceso, en este caso  $i=0$ , donde  $a(ii)$  es  $a(00)$ .
  - b) Deben dividir cada coeficiente de toda la ecuación  $i$  por el valor  $a(ii)$  de manera que el término  $a(ii)$  sea 1.
  - c) Deben recorrer todas las ecuaciones desde  $k=0$  hasta  $k=3$  evitando  $k == i$
  - d) Para cada ecuación  $k$  deben:
    - Multiplicar cada término de la ecuación  $i$  por  $-a(ki)$  de manera que  $a(ii)$  es el negativo de  $a(ki)$
    - Sumar cada término de la ecuación  $i$  a cada termino de la ecuación  $k$ .
  - e) Al terminar de recorrer todas las ecuaciones  $k$  la columna  $a(ii)$  tendrá un valor diferente de cero y los  $a(ji)$  para  $j \neq i$  serán cero.
  - f) Deben dividir cada coeficiente de la ecuación  $i$  por  $a(ii)$  de manera que  $a(ii)$  sea 1.
  - g) Deben tomar el siguiente valor de  $i$  esto es  $i++$  y repetir desde el punto b.

4. Una vez que recorren toda las ecuaciones i deben tener una diagonal de unos en a(00), a(11),a(22) y a(33) de manera que los valores a(04),a(14),a(24) y a(34) son las soluciones.
5. Si alguna línea da cero (todos los coeficientes son cero) las ecuaciones son linealmente dependientes.
6. Deben indicar la condición Linealmente Dependientes (LD) y terminar el programa.

## Planteamiento programático

### Parte I

Deben resolver el sistema de ecuaciones para grado 4 con el siguiente procedimiento:

1. Deben definir una clase **`_ecuacion_gaus_base`**.
2. Deben definir en forma privada un vector de tipo double de 5 elementos que contiene las ecuaciones similar al usado en el ejemplo de solución con determinantes.
3. Deben asignarle el operador `double& operator[](int)` para acceder a cada elemento interno de **`_ecuacion_gaus_base`**.
4. Deben definir una clase **`_ecuaciones4x4_gaus`** que realice una composición de la ecuación **`_ecuacion_gaus_base`**.
5. Deben resolver el algoritmo descrito anteriormente.
6. Utilizar el archivo main correspondiente adjunto a este documento para resolver el programa.
7. Validar los resultados.

### Parte II

Deben modificar el programa anterior de manera que puedan resolver un sistema de 4 ecuaciones para el tipo T, donde T puede ser double o `_complex` de la siguiente forma:

1. Modificar el código anterior de manera que `_ecuacion_gaus_base` y `_ecuacion4x4_gaus` puedan operar sobre `template <typename T>`
2. Recuerden que todo el código pasa al archivo h.
3. Probar el archivo main correspondiente entregado adjunto a este documento.

## **Metodología de entrega.**

La solución de esta tarea debe realizarse a través de las cuentas individuales mediante la plataforma GIT con la que el estudiante cuenta. La solución es personal y no deben compartir la solución con terceros. En caso de que se detecten tareas iguales. La nota individual será el cociente de dividir la nota final entre las tareas iguales.

Procedimiento de entrega:

1. Deben crear una “rama” en el GIT propio con el nombre “Tarea02\_<carne>”.
2. Deben hacer como mínimo un “Commit” de avance antes de la clase de los días de clase (antes de las 6:00 p.m.). Se pueden hacer más “commits” de requerirse.
3. El objetivo es poder darle trazabilidad detallada al avance del estudiante en la solución de las tareas.
4. Deben indicar detalladamente en cada commit, las tareas realizadas entre cada entrega: archivos incluidos, funcionalidad y descripción de la parte algorítmica. Esto evita el tener que escribir un informe sobre el desarrollo.

### **Nota:**

Esta tarea tiene un puntaje de 10%. 5% en avances de GIT y 5% en funcionamiento adecuado de las tareas pedidas.

Fecha de entrega: viernes 7 de mayo antes de las 9:00 p.m.