

# ACH2033 - Matrizes, Vetores e Geometria Analítica

## Exercício-Programa 1

### Eliminação gaussiana e eliminação de Gauss-Jordan

#### Introdução

A *eliminação gaussiana* e a *eliminação de Gauss-Jordan* são dois processos que transformam uma matriz a partir de três operações elementares:

- Troca da posição de duas linhas.
- Multiplicação de uma linha por um valor escalar.
- Soma de uma linha por outra multiplicada por um escalar, com substituição da primeira linha pelo resultado.

A *eliminação gaussiana* tem por objetivo colocar a matriz na chamada *forma escalonada*. Segundo [1], nesta forma a matriz apresenta as seguintes propriedades:

- A entrada não nula mais à esquerda de cada linha possui o valor igual a 1 (valor chamado de pivô da linha).
- O pivô da linha inferior ocorre sempre à direita do pivô da linha superior.
- Linhas nulas são agrupadas na porção inferior da matriz

Uma consequência direta destas propriedades é que todas as entradas de uma coluna abaixo do pivô terão valor igual a zero. Além disso, se a matriz em questão for quadrada e não houver nenhuma linha nula nesta matriz, a matriz será uma matriz triangular superior, e todos os pivôs estarão localizados na diagonal principal da matriz.

A *eliminação de Gauss-Jordan* dá continuidade ao processo iniciado pela eliminação gaussiana colocando a matriz na chamada *forma escalonada reduzida*. Nesta outra forma, além das propriedades já listadas, também vale a seguinte propriedade:

- Com exceção do pivô, todas as entradas de uma coluna possuem valor igual a zero (ou seja, tanto as entradas de acima quanto as abaixo do pivô são constituídas de zeros).

Observe ainda que, quando a matriz na forma escalonada reduzida for quadrada, sem a presença de linhas nulas, tal matriz corresponderá à matriz identidade.

Dentre as aplicações estudadas da *eliminação gaussiana* e/ou a *eliminação de Gauss-Jordan* podemos citar: a resolução de sistemas de equações lineares, a inversão de matrizes e o cálculo de determinantes.

## Objetivo

O objetivo desta atividade é implementar os dois processos de eliminação, assim como as seguintes funcionalidades baseadas nestes processos:

- Resolução de sistemas lineares
- Inversão de matrizes
- Cálculo do determinante.

Para isso vocês deverão completar o esqueleto de código fornecido para que o programa funcione da forma esperada (vocês poderão optar completar um esqueleto escrito em C ou Java).

Vocês **não podem alterar as declarações das funções/métodos** já existentes e **nem alterar a especificação de entrada/saída considerada pelo programa**. Vocês podem, contudo, criar novas funções/métodos (e classes no caso específico da versão Java) conforme julgarem necessário ou conveniente. O uso de bibliotecas que implementam funcionalidades relacionadas a matrizes e álgebra linear **não é permitido** para a realização desta atividade.

Assumiremos, para este EP, que matrizes envolvidas **serão sempre quadradas**, um requisito quando lidamos com matrizes inversas e cálculo do determinante, mas que implica que os sistemas de equações lineares considerados **terão sempre  $n$  equações e  $n$  incógnitas** (observe que estamos considerando como quadrada apenas a porção da matriz que representa os coeficientes do sistema de equações representado de forma matricial).

## Formato de entrada e saída do programa principal

Cada execução do programa deve executar uma das funcionalidades implementadas: resolver de um sistema de equações lineares expresso na forma matricial; calcular a inversa de uma matriz; ou calcular o determinante de uma matriz. A definição da operação a ser realizada, bem como o conteúdo das matrizes são passados ao programa através da entrada padrão (*stdin*).

De forma mais detalhada, o programa deve executar os seguintes passos:

- 1) Ler uma string que definirá a operação a ser realizada. As strings usadas para representar cada uma das 3 operações são: "resolve", "inverte" e "determinante".
- 2) Ler um valor inteiro que irá determinar a dimensão da matriz quadrada envolvida na operação.

3) Ler o conteúdo da matriz:

- a) Quando a operação a ser realizada for o cálculo da inversa ou do determinante, o conteúdo da matriz é composto por **n linhas, cada uma com n valores** do tipo double.
- b) Quando a operação a ser realizada for a resolução de um sistema de equações, o conteúdo da matriz corresponde à representação matricial do sistema, sendo composto por **n linhas, cada uma com (n + 1) valores** do tipo double. Observem que, nesta representação matricial, a sub matriz à esquerda de tamanho  $n \times n$  representa os coeficientes do sistema (que são multiplicados pelas incógnitas) enquanto a última coluna representa os valores localizados à direita nas igualdades de cada equação.

4) Executar a operação escolhida, e imprimir o resultado na saída padrão:

- a) No caso do cálculo do determinante, deve ser impresso o valor double referente ao determinante calculado.
- b) No caso do cálculo da inversa, deve ser impressa a própria matriz inversa calculada (quando esta for inversível) ou a mensagem “matriz singular” (quando não for inversível).
- c) No caso da resolução de um sistema, deve ser impresso o conjunto solução do sistema de equações quando houver uma solução única. Caso existam diversas soluções deve ser impressa a mensagem “sistema possui diversas soluções”, e quando o sistema não tiver nenhuma solução deve ser impressa a mensagem “sistema sem solução”.

Alguns destes passos já estão implementados ou parcialmente implementados nos esqueletos fornecidos e cabe a vocês completar a função/método `main` para funcionar de acordo com esta especificação (além, é claro, de implementar os processos de eliminação).

Uma dica útil, que irá facilitar a utilização do programa, é criar arquivos de texto contendo toda a informação que gostaríamos de passar ao programa, e redirecionar o conteúdo deste arquivo para a entrada padrão do programa, executando-o da seguinte forma na linha de comando:

```
./ep1 < entrada.txt
```

Um exemplo de arquivo de entrada pode ser visto nas linhas abaixo:

```
resolve
2
4 3 48
2 1 20
```

Para este arquivo de entrada a seguinte saída deve ser gerada:

6.0

8.0

(Outros casos de teste com as saídas esperadas serão fornecidas para auxiliá-los no desenvolvimento deste EP)

## Entrega

Vocês devem escolher uma das versões do código para ser completado (C ou Java), e implementar as funcionalidades necessárias para o adequado funcionamento do programa. A entrega do código fonte completo deverá ser feita através da tarefa criada no eDisciplinas até o dia 13/12 às 23:59. Este EP pode ser feito de forma **individual ou em dupla**. Caso seja feito em dupla, apenas um dos membros precisa fazer o envio. Não se esqueçam de indicar (através de comentários) os nomes e números USP dos membros no início do arquivo fonte. Fiquem à vontade para adicionar ao código comentários que julguem relevantes para o bom entendimento do programa.

Boa diversão! :)

## Referências

[1] Howard Anton Chris Rorres. *Álgebra Linear com Aplicações*. 10ª edição, Bookman, 2012.