



## Objetivos

Escrever um programa que calcule o determinante de uma matriz quadrada  $N \times N$  utilizando o Teorema de Laplace.

## Especificação

1. A primeira linha conterá um número indicando a dimensão da matriz. Como a matriz é quadrada, ambos os números são iguais. Não há limite definido para este número.
2. Em seguida virão os elementos da matriz separados por espaços em branco. Tais elementos podem ser tanto positivos quanto negativos.
3. O cálculo deve ser feito utilizando o Teorema de Laplace, em que o cálculo do determinante de uma matriz de ordem  $N$  é transformado no cálculo de determinantes de matrizes  $N - 1$  recursivamente, até se chegar ao caso base de uma matriz de ordem  $N = 2$  ou  $N = 3$  (sua preferência). Veja a página da Wikipédia para uma definição mais precisa e exemplo de cálculo: [http://pt.wikipedia.org/wiki/Teorema\\_de\\_Laplace](http://pt.wikipedia.org/wiki/Teorema_de_Laplace) (Último Acesso: 10/03/2014, 13:00)
4. Seu programa deve, a cada etapa, utilizar a linha ou coluna com mais zeros para agilizar o cálculo do determinante no caso de matrizes esparsas. Sem aproveitar os zeros de uma matriz, o cálculo do determinante pelo teorema de Laplace tem complexidade  $O(n!)$ , o que o limitaria a matrizes muito pequenas.
5. Ao final do cálculo, o programa deve imprimir o determinante da matriz.

## Exemplo 1

Entrada:

```
3
1 2 3
4 5 6
7 8 9
```

Saída:

```
0
```

## Exemplo 2

Entrada:

```
4
0 -1 0 2
0 0 -9 0
9 3 -1 0
7 0 -3 3
```

Saída:

```
-135
```

## Observações

1. Deve ser utilizada **recursão** no cálculo do determinante.
2. Deverão ser usadas estruturas do tipo `loop`.
3. Deve se utilizar o método proposto (Teorema de Laplace) para toda matriz  $N > 3$ . O programa que utilizar outro método de cálculo de determinante para tais matrizes será considerado **errado**. Obviamente, outros métodos podem ser usados para o(s) caso(s) base em que  $N \geq 3$ .
4. Os laboratórios devem ser compiláveis usando o GCC 4, com a seguinte linha de execução: `gcc -lm -std=c99 -Wall -pedantic -Werror -o main arquivo.c`
5. É obrigatório liberar toda a memória alocada. Os programas que não liberarem a memória alocada serão considerados errados, independentemente da saída correta nos testes.
6. Códigos ilegíveis serão considerados errados. A legibilidade é obtida com indentação correta e coerente, bons nomes de variáveis e funções, bem como boa subdivisão do código em funções auxiliares.