

## SOFTWARE FEITO EM LINGUAGEM C QUE REALIZA CÁLCULOS DE DETERMINANTES DE MATRIZES QUADRÁTICA DE ORDEM ATÉ 3X3.

**Ramon Oliveira Borges dos Santos**

Centro Universitário Salesiano de São Paulo - Unisal  
*Ramonobs98@gmail.com*

**Emiliana Bastos de Amorim**

Centro Universitário Salesiano de São Paulo - Unisal  
*Emiliana.Amorim@gmail.com*

**Resumo.** Atualmente os computadores tornou-se eficiente no cotidiano de alguns pesquisadores, com esse avanço da tecnologia e no desenvolvimento de novas linguagens de programação, notou-se a necessidade de aplicar esse conceito em determinadas pesquisas científicas, para diminuir as chances de erros em cálculos complexos e etc. A computação científica é utilizada abundantemente nas áreas de ciências exatas. Para solução de problemas numéricos, atualmente usa-se o Fortran e a Linguagem C pelo seu alto índice de aceitação entre o público científico por apresentarem boas performances. Este artigo visa promover a aplicação direta da linguagem de programação em um problema matemático na disciplina de álgebra linear, foi desenvolvido um software em linguagem C, que soluciona de maneira eficiente um determinante de matrizes de ordem até três por três utilizando a regra de Sarrus.

**Palavras chave:** Linguagem C. Teorema de Cramer. Programação. Computação Científica.

**Abstract.** Currently the computers became effective in everyday life of some researchers, with the advance of technology and the development of new programming languages, we noted that the need to apply this concept in certain scientific research, to decrease the chances of errors in complex calculations and etc. The scientific computing is used abundantly in the areas of exact sciences. For numerical solution of problems, currently uses the Fortran and C Programming Language by its high rate of acceptance among the general public of science by presenting good performances. This article aims to promote the direct application of the programming language in a math problem in the discipline of linear algebra, software was developed in C language, that solves efficiently a determinant of matrices of order up to three by three using the rule of Sarrus.

**key words:** C Programming Language. Programming. Scientific Computing. Determinant.

### 1. INTRODUÇÃO

A Computação ou Informática, entendida como o corpo de conhecimentos acerca de computadores, sistemas de computação e suas aplicações, engloba aspectos teóricos, experimentais, de modelagem e de projeto (Tim Bell, 2011).

O conceito de um artefato (ou dispositivo, mecanismo) universal que realize computação foi descrito por Alan Turing em 1937, a chamada Máquina de Turing. Basicamente, uma Máquina de Turing descreve um mecanismo formal de computação, ou seja, é possível descrever a computação em si através de um conjunto de regra (Gotardo, 2015).

Foi modelado um software computacional, programado em linguagem C para resolver determinantes de matrizes quadráticas na ordem até 3x3, usufruindo ao máximo de cada particularidade das funções da linguagem, ele se mostrou bastante eficiente uma ferramenta indispensável na resolução de exercícios da disciplina de álgebra linear, pois diminui ao máximo as chances de erro.

## 2. FUNDAMENTAÇÃO TEÓRICA

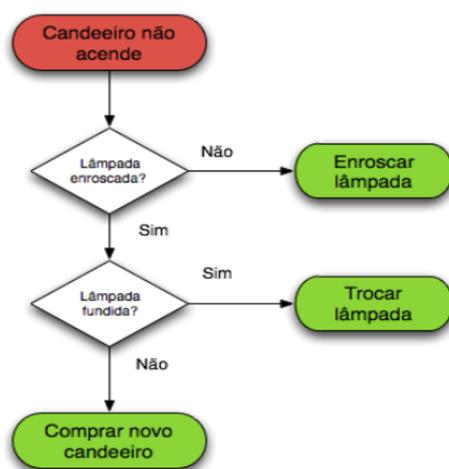
### 2.1 Algoritmo

Usar a lógica é um fator a ser considerado por todos, principalmente pelos profissionais da área de tecnologia de Informação (programadores, analistas de sistemas e suporte), pois no seu cotidiano dentro das organizações é solucionar problemas e atingir os objetivos apresentados por seus usuários com eficiência e eficácia (Manzano, 2005).

Algoritmo é um processo de cálculo matemático ou da descrição sistemática da resolução de um grupo de problemas semelhantes. Pode ser dizer também que são regras formais para obtenção de um resultado ou da solução de um problema, englobando fórmulas de expressões aritméticas. Na área de desenvolvimento de software, é muito comum relacionar a palavra algoritmo com diagrama de bloco (neste caso, seria um algoritmo gráfico), já que muitas fórmulas estão dentro das simbologias de processo para a resolução de um determinado problema (Manzano, 2005).

Pseudocódigo é uma forma genérica de escrever um algoritmo, utilizando uma linguagem simples. Emprega uma linguagem intermediária entre a linguagem natural e uma linguagem de programação. Na Figura 1, tem-se um fluxograma para a reparação da lâmpada, nota-se que se trata de uma sequência de tarefas para a solução definitiva do problema.

Figura 1. Fluxograma de reparação de uma lâmpada



Fonte: Algoritmo

### 2.2 Linguagem C

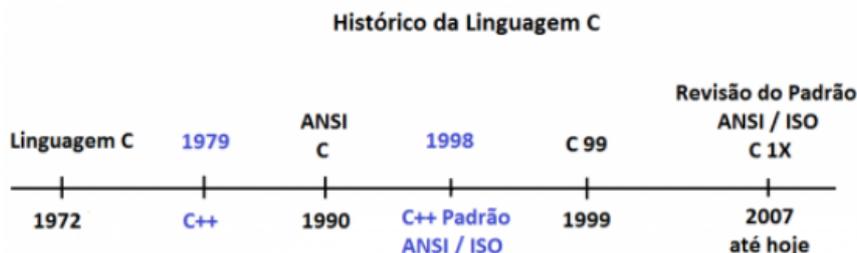
Nos laboratórios da empresa americana: "Bell Labs", nos anos 70, Dennis M. Ritchie e Ken Thompson criaram a linguagem C. Com origem na linguagem B de Thompson, que foi uma evolução da linguagem BCPL. B era a primeira letra de BCPL e C a segunda, portanto os autores acharam que seria mais lógico chamar de linguagem C (Reis, 2015).

A linguagem C, é utilizada no desenvolvimento de sistemas operacionais e em diversos tipos de aplicações. Tem influência direta nas linguagens C++, Java, C#, PHP e JavaScript (Reis, 2015).

A linguagem C é considerada uma linguagem capaz de ser usada para praticamente qualquer tipo de projeto. É uma linguagem portátil, um programa escrito em linguagem C pode ser facilmente usados em qualquer plataforma. Utilizando linguagem C pode-se criar sistemas operacionais, aplicativos de diversos tipos e programar micro controladores.

Rapidamente a linguagem C se tornou uma das linguagens mais populares entre os programadores, pois é poderosa, portátil, flexível e foi desenhada para ser de propósito geral, estruturada, imperativa, procedural e padronizada (Gotardo, 2015).

Figura 2. Histórico da Linguagem C



Fonte: O que é Linguagem C ?

### 2.3 Matrizes

Em uma matriz  $M$ , cada elemento é indicado por  $a_{ij}$ . O índice “ $i$ ” indica a linha e o índice “ $j$ ” a coluna às quais o elemento pertence. Com a convenção de que as linhas sejam numeradas de cima para baixo (de 1 até  $m$ ) e as colunas, da esquerda para direita (de 1 até  $n$ ), uma matriz  $m \times n$  é representada (Iezzi, 1977).

Alguns exemplos de matrizes de ordem quadrática, na imagem a seguir notaremos que pode ser representada de 3 formas diferentes de acordo com a figura 3.

Figura 3. Representação de matrizes

$$\begin{aligned}
 \mathbf{M} &= \begin{bmatrix} \mathbf{a}_{11} & \mathbf{a}_{12} & \dots & \mathbf{a}_{1n} \\ \mathbf{a}_{21} & \mathbf{a}_{22} & \dots & \mathbf{a}_{2n} \\ \dots & \dots & \dots & \dots \\ \mathbf{a}_{m1} & \mathbf{a}_{m2} & \dots & \mathbf{a}_{mn} \end{bmatrix} \text{ ou } \mathbf{M} = \begin{pmatrix} \mathbf{a}_{11} & \mathbf{a}_{12} & \dots & \mathbf{a}_{1n} \\ \mathbf{a}_{21} & \mathbf{a}_{22} & \dots & \mathbf{a}_{2n} \\ \dots & \dots & \dots & \dots \\ \mathbf{a}_{m1} & \mathbf{a}_{m2} & \dots & \mathbf{a}_{mn} \end{pmatrix} \text{ ou} \\
 \mathbf{M} &= \left\| \begin{array}{cccc} \mathbf{a}_{11} & \mathbf{a}_{12} & \dots & \mathbf{a}_{1n} \\ \mathbf{a}_{21} & \mathbf{a}_{22} & \dots & \mathbf{a}_{2n} \\ \dots & \dots & \dots & \dots \\ \mathbf{a}_{m1} & \mathbf{a}_{m2} & \dots & \mathbf{a}_{mn} \end{array} \right\|
 \end{aligned}$$

Fonte: Iezzi, 1977

Uma matriz  $M$  do tipo  $m \times n$  também pode ser indicada por:

$$M = (a_{ij}) ; i \in \{1, 2, 3, \dots, m\} \text{ e } j \in \{1, 2, 3, \dots, n\} \text{ ou simplesmente } M = (a_{ij})_{m \times n} .$$

Matriz quadrada de ordem  $n$  é toda matriz do tipo  $n \times n$ , isto é, uma matriz que tem igual o numero de linhas e de colunas (Iezzi, 1977).

Figura 4. Representação de matrizes quadráticas

$$\left| \begin{array}{cccc} \mathbf{a}_{11} & \mathbf{a}_{12} & \mathbf{a}_{13} & \dots & \mathbf{a}_{1n} \\ \mathbf{a}_{21} & \mathbf{a}_{22} & \mathbf{a}_{23} & \dots & \mathbf{a}_{2n} \\ \mathbf{a}_{31} & \mathbf{a}_{32} & \mathbf{a}_{33} & \dots & \mathbf{a}_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ \mathbf{a}_{n1} & \mathbf{a}_{n2} & \mathbf{a}_{n3} & \dots & \mathbf{a}_{nn} \end{array} \right|$$

Fonte: Iezzi, 1977

Chama-se diagonal principal de uma matriz quadrada de ordem n o conjunto dos elementos que tem os dois índices iguais, isto é:

$$\{a_{ij} \mid i = j\} = \{a_{11}, a_{22}, a_{33}, \dots, a_{nn}\}$$

Chama-se diagonal secundária de uma matriz quadrada de ordem n o conjunto dos elementos que tem soma dos índices a n + 1, isto é:

$$\{a_{ij} \mid i + j = n + 1\} = \{a_{1n}, a_{2, n-1}, a_{3, n-2}, \dots, a_{n1}\}$$

Figura 5. Exemplo de Matriz Quadrática

Exemplos

1° ) A matriz  $M = \begin{vmatrix} 8 & 9 & -7 \\ 6 & 4 & -5 \\ -1 & 2 & 3 \end{vmatrix}$  é quadrada de ordem 3. Sua diagonal principal é  $\{8, 4, 3\}$  e sua diagonal secundária é  $\{-7, 4, -1\}$

Fonte: Iezzi, 1977

### 3. OBJETIVOS

Construir um software que realiza cálculos de determinantes de matrizes quadráticas de ordens 1x1, 2x2 e 3x3. O desenvolvimento foi escrito em linguagem C, pela sua facilidade de entendimento e alto poder de solução e aplicação em diversas áreas da engenharias e computação em geral.

Podemos dizer também que um outro objetivo secundário desse trabalho é despertar o interesse em alunos das diversas áreas do conhecimento, a importância de se conhecer e aprender uma linguagem de programação atualmente, a sua aplicabilidade é infinita, podemos solucionar problemas matemáticos, construir sistemas de estoque ou de gerenciamento de empresas, segurança de sistemas bancários etc. E por fim dizer os benefícios que isso traz ao estudante, um poder de concentração elevado, raciocínio lógico, automatização de processos ou etapas.

### 4. JUSTIFICATIVA

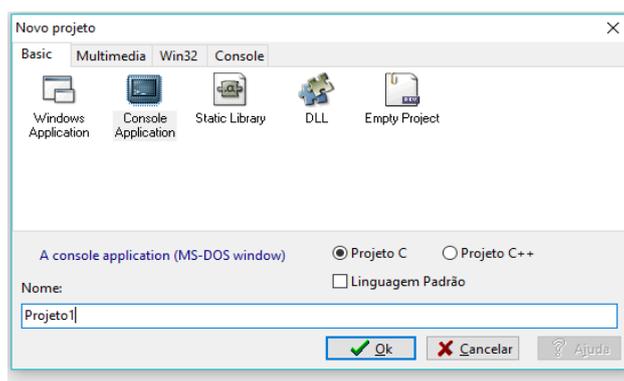
Durante a vida acadêmica de um estudante de engenharia, matemática e física, sendo cursos de ciências exatas extremamente complexos, se nota uma necessidade de utilizar a tecnologia do século XXI a favor do conhecimento e da economia de tempo, durante a disciplina de Álgebra Linear o aluno tem o primeiro contato com matrizes, vale salientar que esses conceitos passados nessa disciplina não se trata de matemática elementar, logo viu-se a necessidade de criar algo que auxilie os alunos a estudar e saber se realizaram essa conta corretamente, logo seu entendimento sobre o assunto terá cada vez mais destaque, também poderá ser utilizada por professores em sala de aula para resolução de problemas durante a aula e na correção de provas, o intuito dessa ideia é disponibilizar gratuitamente aos interessados no software, vale ressaltar que essa aquisição será de grande importância a essas pessoas pois irão adquirir uma ferramenta acessível monetariamente e eficaz sem precisar investir em uma calculadora gráfica ou científica de alta performance.

## 5. METODOLOGIA

### 5.1 Compilador

Durante a confecção do programa foi utilizado o compilador DEV-C++, uma interface gratuita onde podemos escrever a sintaxe da linguagem, que será convertida em linguagem máquina para o computador poder interpreta-la e realizar o que se ordena.

Figura 6. Seleção de um novo projeto

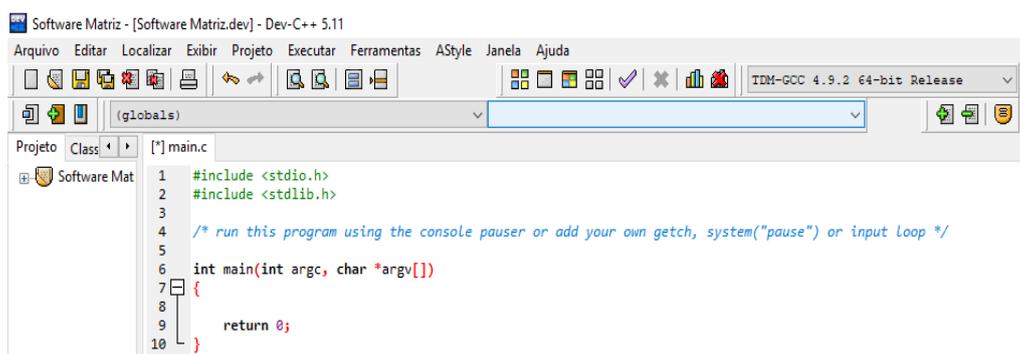


Fonte: Autoria própria

Devemos salientar que na Figura 6, para começar a desenvolver um novo projeto precisamos ir em *Console Application*, escolher a linguagem que desejamos trabalhar, no caso o programa oferece duas a linguagem C e C++, feito isso indicamos que coloque um nome ao seu projeto e clicar no ícone verde com a palavra OK.

Na Figura 7 já se encontra o ambiente pronto para escrever a sintaxe desejada, ela deve ser escrita dentro dos colchetes, devemos ressaltar que é imprescindível escrever a biblioteca necessária para seu projeto, antes de começar a desenvolver. O **return 0**; deve-se encontrar sempre no final no projeto na última linha do código.

Figura 7. Ambiente de desenvolvimento do projeto



Fonte: Autoria própria



O programa de computador, feito em linguagem C que resolve o determinante de matriz de ordem quadrática de 2x2, de acordo com a Figura 10, podemos observar a sintaxe do programa, para atribuição dos valores da matriz foi utilizado um vetor, na linguagem C, o vetor serve para armazenar valores ou informações desejadas para serem utilizada futuramente, utiliza-se vetores para trabalhar com grandes quantidades de números ou caracteres longos.

Figura 10. Programa da Matriz 2x2

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <math.h> // Biblioteca de matemática
4 /* Esse programa realiza o cálculo determinantes de matrizes de ordem 2x2 */
5
6 int main(int argc, char *argv[])
7 {
8     float vetor[4]={0}, determinante; //Declarando variáveis do tipo float
9     int i, resp; //Declarando variáveis do tipo inteiro "int"
10
11     printf("Esse programa calcula determinantes de matrizes quadráticas de valores inteiros sendo sua ordem 2x2 \n"); //Inicialização do programa
12     printf("Exemplo de atribuição dos valores \n \n"); //Exemplo de como atribuir os valores da matriz
13
14     printf("---- \n");
15     printf("-- valor A \t valor B --\n");
16     printf("-- valor C \t valor D --\n");
17     printf("---- \n");
18
19     printf("Atribuição dos valores correspondentes, os valores atribuídos serão descritos linha por linha sendo: \n");
20     printf("O valor A --> Primeiro numero \n");
21     printf("O valor B --> Segundo numero \n");
22     printf("O valor C --> Terceiro numero \n");
23     printf("O valor D --> Quarto numero \n");
24     printf("O valor decimal precisa ser escrito da seguinte forma, separada por ponto ==> . \n exemplo : \n 3/2 --> 1.5 \n 1/2 --> 0.5 \n"); //Exemplo de como atribuir valores decimais na matriz
25     printf("Digite os valores da matriz \n");
26     for( ; ; ) // Loop infinito
27     {
28         do
29         {
30             for (i=0; i<4; i++) //Atribuição dos valores da matriz
31             {
32                 scanf_s("%f", &vetor[i]);
33             }
34             printf("---- \n");
35             printf("-- %.2f \t%.2f--\n", vetor[0], vetor[1]); //Imprimindo a matriz digitada
36             printf("-- %.2f \t%.2f--\n", vetor[2], vetor[3]);
37             printf("---- \n");
38
39             printf("Esta matriz esta correta \n"); //O programa pergunta se a matriz está digitada corretamente
40             printf("Digite 1 ----> sim \n Digite 2----> não \n"); // 1 para sim , 2 para não.
41             printf("Digite sua resposta .....: ");
42             scanf_s("%d", &resp); //resposta armazenada na variável "resp"
43
44             if(resp==1) //Desvio condicional composto "SE" a resposta for 1
45             {
46                 determinante = ((vetor[0]*vetor[3])-(vetor[1]*vetor[2])); //Cálculo do Determinante
47                 printf("Resultado do Determinante .....: %.2f \n \n", determinante); //Imprimindo na tela o valor do determinante
48             }
49             else //Desvio condicional "SE NÃO" for a resposta igual a 1, no caso 2
50             {
51                 printf("Digite sua matriz novamente na forma correta \n"); //Imprime na tela para digitar a matriz novamente
52             }
53
54             while (resp==2); // Se a resposta for 2 ele inicia o comando "do"
55             printf("<===== \n");
56             printf("Continue inserindo valores para calcular resultados de determinantes de matrizes quadráticas com valores inteiros como mostrado no exemplo anterior \n \n");
57             printf("<===== \n");
58         }
59     }
60     return 0; //Fim do programa
61 }
```

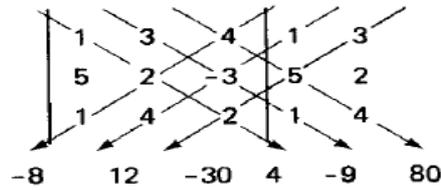
Fonte: Autoria própria

### 5.4 Matriz 3x3

Com matrizes de ordem 3x3, foi utilizado o mesmo princípio do anterior também foi utilizado o princípio de vetores porem com uma quantidade de espaço maior para o armazenamento das variáveis.

Para o cálculo de determinantes de matrizes 3x3, onde consiste em utilizar o método de Sarrus para obtenção do determinante, de acordo com o livro Fundamentos da Matemática Elementar 4, o cálculo é feito de acordo com a Figura 11 e Figura 12. Na “Figura 13” e “Figura 14” podemos notar a programação que foi utilizada.

Figura 11. Método de Sarrus para resolução do determinante



Fonte: Iezzi. 1977

Figura 12 – Método de Sarrus para resolução do determinante

$$\begin{vmatrix} 1 & 3 & 4 \\ 5 & 2 & -3 \\ 1 & 4 & 2 \end{vmatrix} = 4 \cdot -9 + 80 - 8 + 12 - 30 = 49$$

Fonte: Iezzi. 1977

Figura 13. Início do programa, atribuição de valores na matriz

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <math.h> //Declaração da biblioteca de Matemática
4
5 /* Esse programa calcula determinantes de Matrizes 3x3 */
6
7 int main(int argc, char *argv[])
8 {
9     float vetor[9]={0}, determinante; //Declarando Variavel do tipo float
10    int resp, i; // Declarando variavel do tipo inteira
11
12    printf("Esse programa calcula determinantes de matrizes quadraticas com valores inteiros 3x3 \n"); //Inicio do programa
13    printf("Exemplo de atribuição dos valores \n \n"); //Exemplo de atribuição de valores
14
15    printf("---- \n");
16    printf("-- valor A \t valor B \t valor C --\n");
17    printf("-- valor D \t valor E \t valor F --\n");
18    printf("-- valor G \t valor H \t valor I --\n");
19    printf("---- \n \n");
20
21    printf("Atribuição dos valores correspondentes,os valores atribuidos serao descritos linha por linha sendo: \n");
22    printf("O valor A --> Primeiro numero \n");
23    printf("O valor B --> Segundo numero \n");
24    printf("O valor C --> Terceiro numero \n");
25    printf("O valor D --> Quarto numero \n");
26    printf("O valor E --> Quinto numero \n");
27    printf("O valor F --> Sexto numero \n");
28    printf("O valor G --> Setimo numero \n");
29    printf("O valor H --> Oitavo numero \n");
30    printf("O valor I --> Nono numero \n \n");
31    printf("O valor decimal precisa ser escrito da seguinte forma, separada por ponto ==> . \n exemplo : \n 3/2 --> 1.5 \n 1/2-->0.5 \n \n"); //Exemplo de como digitar valor decimal
32    printf("Digite os valores da matriz \n"); //Atribuição dos coeficientes
33

```

Fonte: Autoria própria



Figura 16. Determinante de Matriz 1x1

```
=====>      Esse programa calcula determinantes de matrizes quadraticas 1x1
           Exemplo de atribuicao dos valores
           ----
           -- valor A --
           ----
Atribuicao dos valores correspondentes os valores atribuidos serao descritos linha por linha sendo:
0 valor A --> Primeiro numero
Digite o valor da matriz
3
A matriz esta correta (3.00)
Digite 1 ---> Sim
Digite 2 ---> Nao
Digite.....:1
RESULTADO DO DETERMINANTE.....: 3.00
```

Fonte: Autoria própria

Com a matriz 2x2 os resultados também foram satisfatórios, para o resultado do determinante é necessário realizar a multiplicação da diagonal principal, também é necessário fazer a multiplicação da diagonal secundária, depois efetuar a subtração entre a diagonal principal pela diagonal secundária, como podemos observar o funcionamento do programa e o resultado apresentado Figura 17.

Figura 17. Determinante de Matriz 2x2

```
Esse programa calcula determinantes de matrizes quadraticas de valores inteiros sendo sua ordem 2x2
Exemplo de atribuicao dos valores
----
-- valor A      valor B --
-- valor C      valor D --
----
Atribuicao dos valores correspondentes,os valores atribuidos serao descritos linha por linha sendo:
0 valor A --> Primeiro numero
0 valor B --> Segundo numero
0 valor C --> Terceiro numero
0 valor D --> Quarto numero
0 valor decimal precisa ser escrito da seguinte forma, separada por ponto ==> .
exemplo :
3/2 --> 1.5
1/2-->0.5
Digite os valores da matriz
13
7
11
5
----
-- 13.00      7.00--
-- 11.00      5.00--
----
Esta matriz esta correta
Digite 1 ---> sim
Digite 2---> nao
Digite sua resposta .....: 1
Resultado do Determinante .....: -12.00
```

Fonte: Autoria própria

Logo a matriz 3x3 não é muito diferente o cálculo do determinante, geralmente utiliza-se o método de Sarrus, a princípio o usuário entra com valores da matriz, após digitada o usuário tem a oportunidade de confirmar se digitou os valores corretamente, digitado os valores corretamente na parte inferior da tela será imprimido o resultado, se ele digitar errado, terá uma nova oportunidade para digitar corretamente, esse ciclo será repetido infinitas vezes como podemos observar na Figura 18 e Figura 19.

Figura 18. Atribuição de valores

```
Esse programa calcula determinantes de matrizes quadraticas com valores inteiros 3x3
Exemplo de atribuicao dos valores

----
-- valor A      valor B      valor C --
-- valor D      valor E      valor F --
-- valor G      valor H      valor I --
----

Atribuicao dos valores correspondentes,os valores atribuidos serao descritos linha por linha sendo:
0 valor A --> Primeiro numero
0 valor B --> Segundo numero
0 valor C --> Terceiro numero
0 valor D --> Quarto numero
0 valor E --> Quinto numero
0 valor F --> Sexto numero
0 valor G --> Setimo numero
0 valor H --> Oitavo numero
0 valor I --> Nono numero

0 valor decimal precisa ser escrito da seguinte forma, separada por ponto ==> .
exemplo :
3/2 --> 1.5
1/2-->0.5
```

Fonte: Autoria própria

Figura 19. Determinante da Matriz 3x3

```
Digite os valores da matriz
1
5
-2
8
3
0
4
-1
2

A Matriz esta digitada corretamente

----
-- 1.000      5.000      -2.000 --
-- 8.000      3.000      0.000 --
-- 4.000      -1.000     2.000 --
----

Esta Matriz esta correta
Digite 1 ---> sim
Digite 2----> nao

Digite a sua resposta.....:1
RESULTADO DO DETERMINANTE.....: -34.000 ←
```

Fonte: Autoria Própria

## 7. CONCLUSÃO

Atualmente o engenheiro precisa dispor de tecnologias para a solução de problemas, ao se deparar com situações no meio industrial ele precisa mentalizar como solucionar da maneira sucinta e eficaz. Problemas estes que muitas das vezes necessitam de supercomputadores ou software dedicados a solucionar aquele empecilho, um grande exemplo é o supercomputador do Instituto Nacional de Pesquisas Espaciais o (INPE), o computador conhecido como Tupã oferece a meteorologia de todo Brasil e toda América Latina, o mesmo opera em Cachoeira Paulista – SP no CPTEC – INPE.

Podemos concluir que o software mostrou um ótimo desempenho, apresentando ao usuário os resultados corretos, ele torna uma excelente ferramenta para ser utilizada na disciplina de Álgebra Linear onde é abordado o tema de matrizes, com operações extensas as chances de erros se torna consideravelmente grande. Logo esse software será distribuído gratuitamente aos estudantes de engenharia, matemática, física e demais estudante de áreas afins e professores interessados utilizar esse recurso, com intuito de ajudar o discente e docente a estudar e com isso poder comparar se fez corretamente seu exercício.

## REFERÊNCIAS

**Algoritmo.** Disponível em: <https://pt.wikipedia.org/wiki/Algoritmo> - Acessado no domingo 11/03/2018.

Gotardo, R. A. (2015). **Liguagem de Programação I**. Rio de Janeiro: SESES.

Iezzi, G, Hazzan. S (1977). **Fundamentos da Matemática Elementar 4**. São Paulo: Atual Editora.

José Augusto N. G. Manzano, J. F. (2005). **Algoritmos Lógica para Desenvolvimento de Programação de Computadores**. Érica Ltda.

**O que é Linguagem C?**. Disponível em: <http://linguagemc.com.br/o-que-e-linguagem-c/> - Acessado no domingo 11/03/2018 às 22:35

Reis, A. H. (2015). **Programando em linguagem C e C++**. Santa Cruz do Rio Pardo: Editora Viena.

Tim Bell, I. H. (2011). **Computer Science Unplugged: Ensinando**. Bahia: FAPESB.