

UNIVERSIDADE DE SÃO PAULO

SSC0600

INTRODUÇÃO A CIÊNCIA DA COMPUTAÇÃO I

**RELATÓRIO TRABALHO III
MATRIZ ESPARSA**

Alunos:

Gabriel Santos Nicolau - 10684600

Vinícius Ribeiro da Silva - 10828141

São Carlos, 14 de Junho de 2018

1.INTRODUÇÃO

P = 0: Gabriel Santos Nicolau - 10684600

P = 1: Vinícius Ribeiro da Silva - 10828141

O trabalho consiste em implementar um programa na linguagem C para que se possa criar e manipular uma matriz esparsa. As opções possíveis são:

- Criar matriz;
- Consultar valor de uma posição;
- Atribuir valor a uma posição;
- Calcular a soma de uma linha ou coluna;
- Calcular o determinante da matriz caso essa seja quadrada.

2. DESCRIÇÃO DO PROJETO

2.1 FUNCIONAMENTO DO PROGRAMA

O programa funciona utilizando a alocação dinâmica de memória, onde são criadas structs auto referenciadas, com ponteiros em seu interior apontando para outras structs do mesmo tipo. Dessa maneira, cria-se uma lista encadeada que guarda os valores diferentes de "0" da matriz junto com sua posição na mesma.

Quando se cria uma matriz, todos os elementos iniciais são iniciados como "0". Ao se inserir um valor, apenas essa informação será guardada, evitando o uso desnecessário de memória. Assim que iniciado, o programa cria automaticamente uma matriz inicial, solicitando ao usuário a quantidade de linhas e colunas. Além disso, será possível atribuir valores a posições diretamente, caso o usuário deseje.

Outra struct chamada "cabecalho" armazena o número de linhas e colunas na matriz, auxiliando sua manipulação pelas funções do programa.

O determinante é calculado através de escalonamento, ou método de Gauss, pelo Teorema de Jacobi, em que se obtém uma matriz triangular. Assim, o determinante será dado pelo produto dos elementos da diagonal principal.

2.2 AMBIENTE DE DESENVOLVIMENTO

O projeto foi desenvolvido no Windows 10 x64 como também no Ubuntu 16.04 LTS. Foram utilizados no Windows:

→ Visual Studio 2017

→ Code Blocks 17.12

→ Dev-C++ 5.10

No Ubuntu foi utilizado apenas o editor de texto Kate.

2.3 COMPILAÇÃO

Foram utilizados os compiladores Microsoft Visual C e MinGW GCC 6.3.0 no Windows, já no Ubuntu foi usado o compilador gcc 5.4.0.

2.4 CÓDIGOS FONTE

- Os arquivos fonte utilizados foram: main.c,
- As bibliotecas padrão utilizadas foram: stdio.h, stdlib.h, stdbool.h, limits.h e float.h

3. TUTORIAL

3.1 COMPILAÇÃO

Para o linux, instale com o compilador gcc (caso já não esteja instalado), abra terminal na pasta onde está localizado os códigos fonte e execute o comando:

\$ make

Ou caso não deseje utilizar o makefile é possível compilar utilizando o comando abaixo:

\$ gcc main.c -o main -lm

Para executar o programa utilize do comando

\$./main

No windows é possível abrir, compilar e executar o arquivo com programas como Codeblocks, DevC++, Visual Studio Code, Visual Studio ou similares. Desde que se tome o cuidado para que o programa usado reconheça todos os arquivos fonte, seja, por exemplo, colocando eles em um mesmo projeto ou configurando o programa para aceitar diferentes arquivos fonte.

3.2 EXECUÇÃO DO PROGRAMA

Ao executar o programa será a tela inicial com o Título do programa e o nome de seus autores.



Figura 1. Tela inicial do programa. (Elaborada pelos autores)

Após o pressionamento do botão ENTER será solicitada a criação de uma matriz inicial.

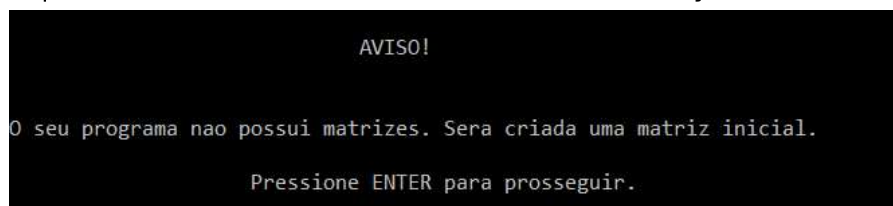


Figura 2. Aviso de não existência de matriz inicial

Após, o usuário deverá digitar os valores da matriz, bem como atribuição de valores, caso deseje:

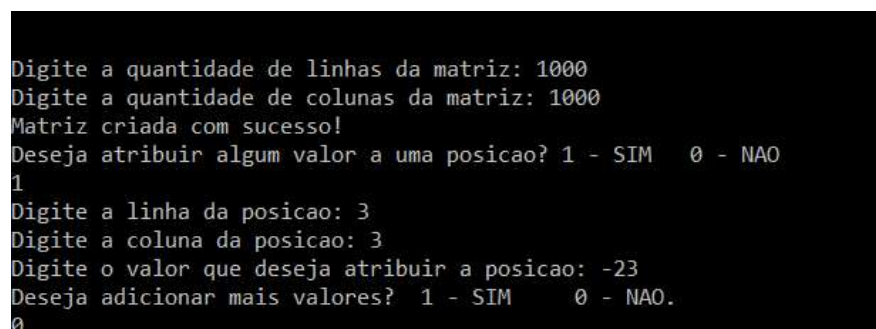


Figura 3. Inserção de valores na matriz

Ao finalizar a atribuição de valores ou apenas a criação da matriz, será exibido o menu principal.

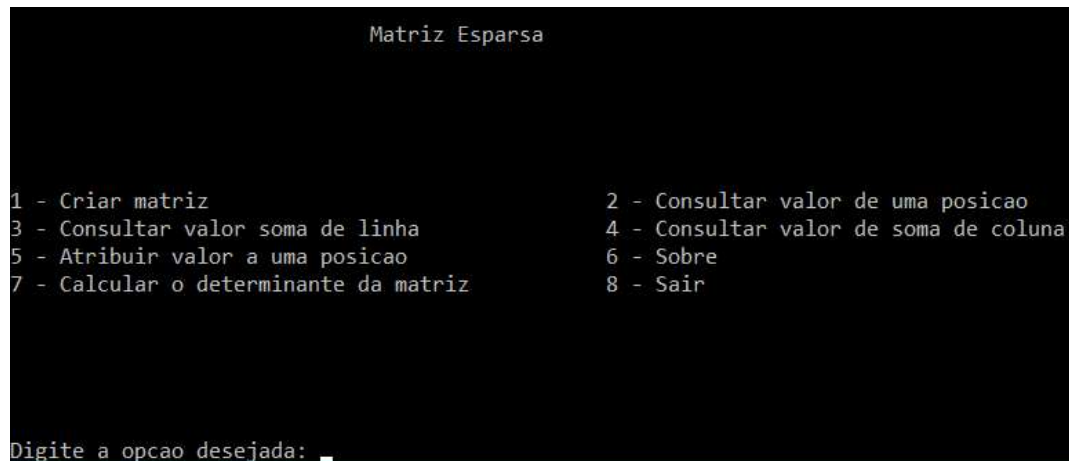


Figura 4. Menu principal.

Aqui aparecem 8 opções:

1. Criar a matriz;
2. Exibir o valor de uma determinada posição
3. Exibir o valor da soma dos elementos de uma linha
4. Exibir o valor da soma dos elementos de uma coluna
5. Atribuir valor a uma posição de escolha;
6. Exibir às informações dos autores e do programa;
7. Calcular o valor do determinante da matriz;
8. Sair do programa.

3.2.1 Criar matriz

Será solicitada a quantidade de linhas e colunas da nova matriz. É possível perceber que ocorre o tratamento de erros, pois o programa impede o usuário de inserir valores negativos ou outros caracteres que não sejam números.

```
Digite a quantidade de linhas da matriz: -1
Valor invalido. Digite algo entre 1 e 2147483647.
Digite novamente: valor
Entrada invalida. Digite novamente: 1000
Digite a quantidade de colunas da matriz: -1
Valor invalido. Digite algo entre 1 e 2147483647.
Digite novamente: valor
Entrada invalida. Digite novamente: 0
Valor invalido. Digite algo entre 1 e 2147483647.
Digite novamente: 1000
Matriz criada com sucesso!
Deseja atribuir algum valor a uma posicao? 1 - SIM  0 - NAO
```

Figura 5. Tela de criação de uma nova matriz

Ademais, o programa possibilita inserir valores a posições sem recorrer ao menu principal, caso desejado.

3.2.2 Consultar valores

Será pedida a linha e a coluna da posição ao usuário. Os números serão limitados ao tamanho da matriz, além disso serão impedidas entradas inválidas como strings ou valores negativos

```
Digite a opcao desejada: 2
Digite o valor da linha que deseja consultar: -1
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 10000
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: linha
Entrada invalida. Digite novamente: 1
Digite o valor da coluna que deseja consultar: coluna
Entrada invalida. Digite novamente: -3
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 1
O valor da posicao (1,1) e: 23.0000

Pressione ENTER para voltar ao menu principal
```

Figura 6. Tela de consulta de valores numa matriz 4x4

3.2.3 Consultar valor de soma de linha

Será pedida apenas a linha que se deseja calcular a soma. Novamente ocorre o tratamento de erros.

```
Digite a opcao desejada: 3
Digite a linha que deseja consultar: -1
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 1000
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: linha
Entrada invalida. Digite novamente: 1
A soma da linha 1 e: 23.0000

Pressione ENTER para voltar ao menu principal
```

Figura 7. Consulta de soma de linha com tratamento de erros

3.2.4 Consultar valor de soma de coluna

O usuário deverá digitar a coluna que deseja obter a soma.

```

Digite a opcao desejada: 4
Digite a coluna que deseja consultar: -1
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 123
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: coluna
Entrada invalida. Digite novamente: 1
A soma da coluna 1 e: 23.0000

Pressione ENTER para voltar ao menu principal

```

Figura 7. Consulta de soma de coluna com tratamento de erros

3.2.5 Atribuir valor a uma posição

O usuário deverá inserir a posição do novo valor, primeiro deverá digitar a linha e depois a coluna.

```

Digite a opcao desejada: 5
Digite a linha da posicao: -2
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 123
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: linha
Entrada invalida. Digite novamente: 1
Digite a coluna da posicao: -23
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: coluna
Entrada invalida. Digite novamente: 123333
Valor invalido. Digite algo entre 0 e 3.
Digite novamente: 2
Digite o valor que deseja atribuir a posicao: 25
Deseja adicionar mais valores? 1 - SIM 0 - NAO.

```

Figura 8. Tela de atribuição de valores com tratamento de erros

O programa considera enganos do usuário, pois caso a posição já possua algum valor prévio é perguntado ao usuário se deseja prosseguir com a operação.

```

Digite a linha da posicao: 1
Digite a coluna da posicao: 2
Digite o valor que deseja atribuir a posicao: 25
Elemento na posicao (1,2) ja cadastrado
Deseja cancelar a operacao (-1), tentar novamente (0) ou sobreescrever o elemento (1).

```

Figura 9. Aviso de sobreposição de valores

3.2.6 Sobre

O menu Sobre é composto de três telas separadas por informações a respeito do(s):

- Projeto;
- Autores.

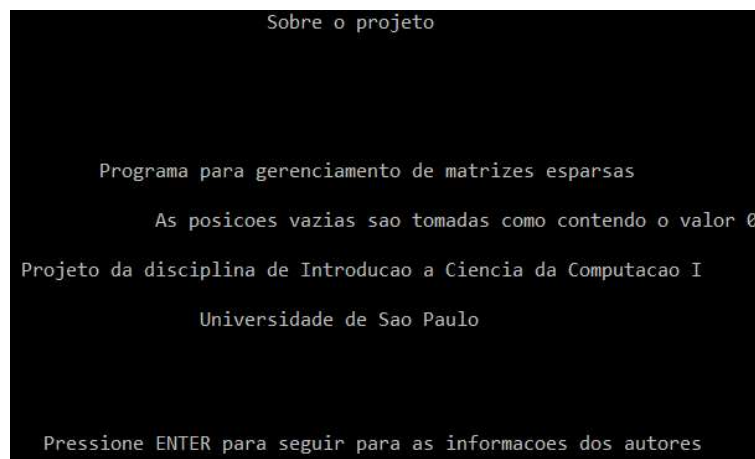


Figura 10. Tela de informações do projeto



Figura 11. Tela de informações dos autores

3.2.7 Calcular o determinante da matriz

O programa faz o cálculo e exibe o valor; caso a matriz não seja quadrada, em que não existe determinante, é exibida na tela a mensagem ao usuário.

Além disso, é possível escolher a opção de acompanhar o passo a passo do cálculo do determinante através do método de Gauss, respeitando, é claro, as condições (ser de 3x3 a 13x13). O programa exibirá a matriz enquanto o usuário pressiona ENTER para ir ao próximo passo.


```

Voce escolheu a opcao de calcular o determinante.
Caso sua matriz seja quadrada e esteja entre 3x3 e 13x13, sera possivel acompanhar o passo a passo.
Deseja acompanhar o passo a passo do metodo? 1 - SIM  0 - NAO
1
Matriz original
000.0  000.0  000.0
000.0  000.0  000.0

Pressione ENTER para prosseguir

Essa matriz nao e quadrada, nao existe determinante.

Pressione ENTER para voltar ao menu principal

```

Figura 12. Mensagem de erro para matriz não quadrada

```

Voce escolheu a opcao de calcular o determinante.
Caso sua matriz seja quadrada e esteja entre 3x3 e 13x13, sera possivel acompanhar o passo a passo.
Deseja acompanhar o passo a passo do metodo? 1 - SIM  0 - NAO
Matriz original
000.0  -005.0  008.0  006.0
001.0  -002.0  001.0  000.0
002.0  -001.0  -005.0  007.0
000.0  002.0  003.0  002.0

Pressione ENTER para prosseguir

Trocando linha 0 com a linha 1
001.0  -002.0  001.0  000.0
000.0  -005.0  008.0  006.0
002.0  -001.0  -005.0  007.0
000.0  002.0  003.0  002.0

```

Figura 13. Início do passo a passo

```

Pressione ENTER para prosseguir

Somando na linha 3, a linha 2 multiplicada por 31.00
001.0  -002.0  001.0  000.0
000.0  -005.0  008.0  006.0
000.0  000.0  011.0  -053.0
000.0  000.0  000.0  -1885.0

Pressione ENTER para prosseguir

Pode-se observar que aqui possuímos uma matriz triangular, em que os elementos abaixo da diagonal principal tem valor 0
Assim, o valor do determinante e o produto dos elementos da diagonal principal

O determinante da sua matriz e: -377.000000

Pressione ENTER para voltar ao menu principal

```

Figura 14. Fim do passo a passo com o resultado do determinante

4. OUTRAS INFORMAÇÕES

4.1 LIMITAÇÕES

- O número máximo de linhas e colunas é limitado pelo tamanho de um **int**;
- O número máximo de um valor armazenado na matriz é limitado pelo tamanho de um **double**.

4.2 Lista encadeada

Como citado anteriormente, o programa usa uma lista encadeada para acessar os valores da matriz. Elas possuem a vantagem de não precisar de um número definido de elementos,, ou seja, a lista aumenta à medida que o usuário precisa incluir um novo elemento nela.

As listas encadeadas são uma das estruturas de dados mais utilizadas no desenvolvimento de programas. São recursos que utilizam mecanismos de acesso encadeado, que armazenam seus dados somente quando necessário, fazendo a liberação da memória quando não mais utilizada. (MADEIRA; SIMÕES; MARTINS, 2004).

No programa, cada valor da matriz fica salvo dentro de uma *struct*, que contém a posição (linha e coluna) e o valor do elemento.

5. Referências

MADEIRA, Maicon Francisco; SIMÕES, Priscyla Waleska Tagino de Azevedo; MARTINS, Paulo João. **ODIN: Ambiente Web de Apoio ao Ensino de Estruturas de Dados Lista Encadeada**. 2004. Disponível em: <<http://periodicos.unesc.net/sulcomp/article/view/800>>. Acesso em: 10 jun. 2018.