

UNIVERSIDADE DE SÃO PAULO
ENGENHARIA DE COMPUTAÇÃO 2018
INTRODUÇÃO À CIÊNCIA DE COMPUTAÇÃO I

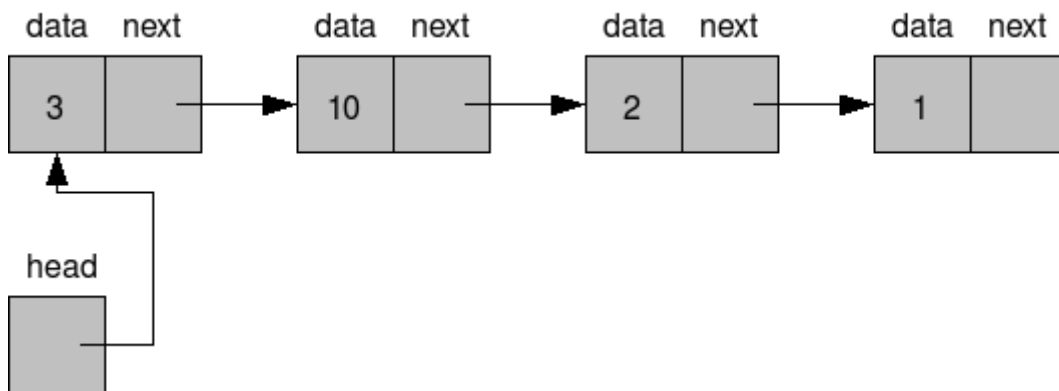
VICTOR FELIPE DOMINGUES DO AMARAL
NUSP:10696506

MATRIZ ESPARSA

INTRODUÇÃO

O trabalho 3 consiste em implementar um programa que permite ao usuário realizar operações em uma matriz do tipo Esparsa. Uma matriz Esparsa é uma matriz em que a maioria dos elementos têm valor nulo. Este tipo de matriz é muito utilizado na computação para estruturar dados. São operações que o usuário pode realizar no programa: criar matriz, cadastrar campo, consultar campo, soma de linhas, soma de colunas e calcular determinante.

O modo como se organiza a matriz é através de uma **lista encadeada**, uma estrutura que aponta para a próxima posição. Desta maneira, não é necessário alocar todas as posições da matriz, apenas algumas. Isso diminui drasticamente o uso da memória. Abaixo há um exemplo de uma lista.



O cálculo do determinante foi feito utilizando o **método do escalonamento de Gauss**, que visa transformar a matriz em uma matriz triangular. Uma matriz triangular possui a propriedade de que seu determinante corresponde ao produto da diagonal principal. No método de Gauss, quando trocamos uma linha pela outra, o determinante tem seu sinal trocado. Quando somamos a uma linha uma outra linha multiplicada por um escalar γ , o determinante da matriz não se altera. A partir destes princípios, podemos transformar uma matriz qualquer em uma matriz triangular. Há alguns métodos para se calcular o determinante de uma matriz, como o de Laplace, mas estes métodos demandam de muito processamento e são inviáveis para matrizes de ordem grande.

Matriz triangular:

$$A = \begin{bmatrix} 5 & 4 & 7 \\ 0 & 3 & 4 \\ 0 & 0 & 6 \end{bmatrix}$$

DESCRIÇÃO DO PROJETO

Ambiente de desenvolvimento:

O ambiente de desenvolvimento utilizado foi a IDE CodeBlocks 17.12, num sistema operacional Windows 64 bits.

Compilador:

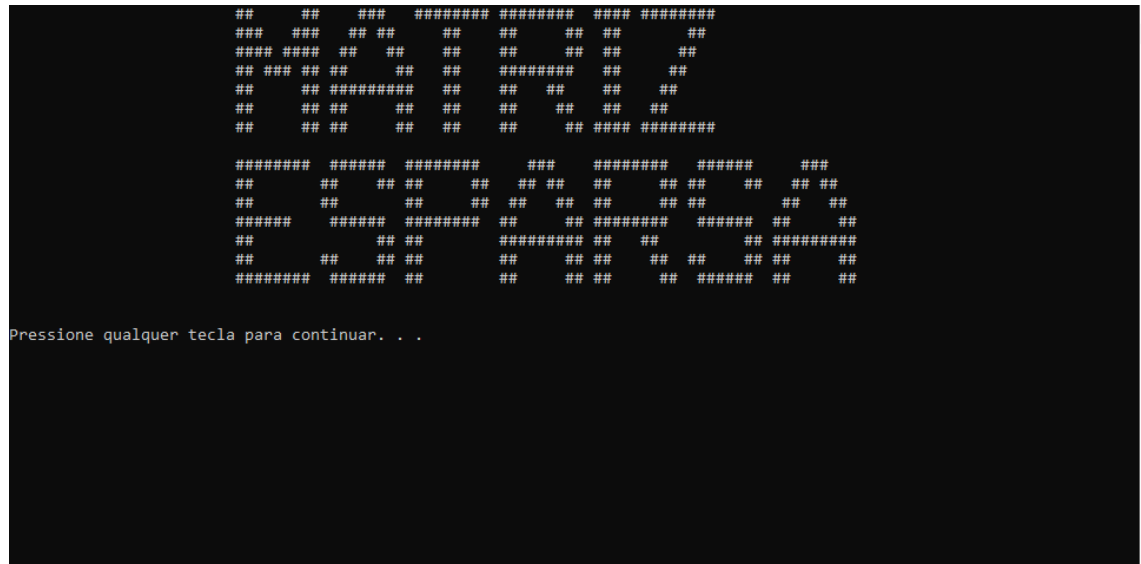
O compilador utilizado no desenvolvimento do programa foi o GCC 64 bits.

Código fonte:

O código fonte se encontra no arquivo main.c, importando os seguintes cabeçalhos: stdio.h, stdlib.h, Windows.h e locale.h e funções.h (declaração das funções).

Entradas Exemplo:

Ao abrir o programa, o usuário entra em uma tela de apresentação do trabalho.



Posteriormente, o programa pede ao usuário para cadastrar a matriz. Os valores aceitos para linhas e colunas são do tipo inteiro positivo. Cabe enfatizar que a matriz possuirá linha e coluna 0. Portanto, quando o usuário insere um valor n de linhas ou colunas, o programa criará n linha ou colunas, sendo a linha 0 a primeira linha.

```
Digite a quantidade de linhas da matriz  
5  
Digite a quantidade de colunas da matriz
```

Caso o usuário digite uma entrada inválida, o programa apontará erro e pedirá para inseri-lo novamente.

```
Digite a quantidade de linhas da matriz  
5  
Digite a quantidade de colunas da matriz  
-9  
Entrada inválida. Digite novamente.
```

Após o cadastro da matriz, o usuário é submetido ao menu do programa:

```
1 -- Inserir valor
2 -- Calcular soma de uma linha
3 -- Calcular soma de uma coluna
4 -- Consultar campo
5 -- Calcular Determinante e encerrar
6 -- Excluir matriz
```

Caso o usuário digite alguma entrada errada, ele apontará o erro.

```
1 -- Inserir valor
2 -- Calcular soma de uma linha
3 -- Calcular soma de uma coluna
4 -- Consultar campo
5 -- Calcular Determinante e encerrar
6 -- Excluir matriz
8

Digite uma entrada correta.
Pressione qualquer tecla para continuar. . .
```

Quando o usuário digitar 1, o programa o levará para o menu de inserção de um campo. Após a inserção, o usuário retornará ao menu principal.

```
Digite o valor:  
5,8  
Digite a linha:  
0  
Digite a coluna:  
3  
Cadastrado com sucesso!  
Pressione qualquer tecla para continuar. . .
```

Caso o usuário digite um valor inválido, como por exemplo uma linha ou coluna que extrapole o limite da matriz, o programa enviará uma mensagem de erro, pedindo para o usuário inserir novamente.

```
Digite o valor:  
9  
Digite a linha:  
9  
Valor incorreto. Digite novamente.
```

Caso o campo já esteja cadastrado, o programa informará ao usuário, perguntando se o usuário deseja sobrescrever o campo.

```
Digite o valor:
5
Digite a linha:
0
Digite a coluna:
0
Campo já cadastrado, deseja sobrescrever?
1 - SIM
QUALQUER VALOR - NAO
```

Voltando ao menu, quando o usuário entrar na função de consultar soma de linha(opção 2), ele será redirecionado a esta tela, onde ele irá inserir a linha em questão:

```
Digite a linha:0
soma da linha: 3,000000
Pressione qualquer tecla para continuar. . .
```

(Esta função e a seguinte também têm o tratamento de entrada inválida, como, por exemplo, um número de linha maior do que a quantidade de linhas da matriz

O mesmo é válido para a soma de coluna:

```
Digite a coluna:
0
soma da coluna: 3,000000
Pressione qualquer tecla para continuar. . .
```

A função consultar posição tem a seguinte característica abaixo, sendo que o usuário insere a linha e coluna a ser consultada:

```
Digite a linha do elemento :
0
Digite a coluna do elemento:
0
O valor do campo 0x0 é : 3,000000
Pressione qualquer tecla para continuar. . .
```

(Esta função também possui o mesmo tratamento de erro das anteriores)

A função calcular “determinante e encerrar” irá ler os campos já registrados e calcular o determinante com base neles, sendo que as posições não registradas têm como valor nulo por padrão. Após o cálculo do determinante, ela irá excluir a matriz e voltar a tela de cadastro de uma nova matriz

```
1 -- Inserir valor
2 -- Calcular soma de uma linha
3 -- Calcular soma de uma coluna
4 -- Consultar campo
5 -- Calcular Determinante e encerrar
6 -- Excluir matriz
5
O valor do determinante é: 99,000
Pressione qualquer tecla para continuar. . .
```

Quando o usuário opta pela opção excluir matriz, o programa exclui a matriz, retornando à tela de cadastro de uma nova matriz.

```
1 -- Inserir valor
2 -- Calcular soma de uma linha
3 -- Calcular soma de uma coluna
4 -- Consultar campo
6 -- Excluir matriz
6
Matriz excluída com sucesso!
Pressione qualquer tecla para continuar. . .
```

LIMITAÇÕES E BUGS

Para que a função calcule o determinante, é necessário trocar alguns valores da matriz. Isso faz com que a matriz original seja perdida. A solução para esta situação seria criar uma cópia da matriz principal para efetuar a operação. Entretanto, duplicar a matriz para efetuar a operação demanda um grande gasto de memória, caso a matriz tenha proporções significativamente grandes.

O programa utiliza de alguns recursos que somente funcionam no sistema operacional Windows, como a limpeza do buffer do teclado e a limpeza de tela. Portanto, é necessário compilá-lo neste sistema.