



Universidade de São Paulo

SSC0600 Introdução a Ciência da Computação I

Simulador de matriz esparsa

São Carlos

2018

Gabriel Soares Gama

Relatório elaborado pelo aluno Gabriel Soares
Gama, matriculado no curso de Engenharia de
Computação, na Universidade de São
Paulo USP do campus de São Carlos. O
trabalho foi orientado pelo Professor
Doutor Adenilso da Silva Simão.

SÃO CARLOS

15/06/2018

INTRODUÇÃO

Este programa é uma implementação em c de um simulador de matriz esparsa que permite a manipulação da mesma.

O usuário pode criar e remover a matriz, inserir, modificar e visualizar os elementos inseridos na mesma.

MEMBROS DA EQUIPE

Gabriel Soares Gama – 10716511 p = 0

DESCRIÇÃO DO PROJETO

O programa foi desenvolvido a partir do editor de texto Visual Studio Code no Ubuntu 18.04 LTS (64 bits).

O compilador utilizado foi o GNU Compiler Collection (GCC).

O arquivo `matriz.c` contém o código fonte e é necessário para compilar o programa no windows, no linux o arquivo correspondente é o `matriz.c`. Para a compilar o código não é necessário nenhum parâmetro extra de compilação.

O programa se inicia com um menu principal com 9 opções sendo elas:

“A”, “B”, “C”, “D”, “E”, “F”, “G”, “H”, “I” (o programa também aceita letras minúsculas).

O usuário deve primeiramente criar uma matriz (opção “B”) para poder realizar as outras operações presentes, com exceção de “Fechar programa”. Após isso, o programa irá pedir para o usuário digitar os limites da matriz, ou seja, o numero de colunas e de linhas.

Após criar uma matriz, o usuário pode adicionar elementos a mesma ou removê-la. Caso deseje adicionar elementos na matriz, o usuário deve digitar “c” ou “C” e é necessário fornecer o número de elementos que deseja adicionar à matriz. Depois de fornecer a quantidade de dados que serão adicionados na matriz, o usuário deve fornecer o valor, a linha e a coluna para cada número que irá adicionar. Essa ação pode ser repetida diversas vezes, caso respeite os limites das matrizes, como o número máximo de espaços vazios (não adicionados pelo usuário) na matriz.

Se o usuário escolher a opção de modificar os elementos na matriz, digitando “d” ou “D”, este deve digitar a quantidade de elementos que deseja modificar e a linha e a coluna de um elemento já adicionado anteriormente pelo usuário. Essa ação deve ser repetida até que o número de modificações restantes se torne zero.

As opções “E” e “F” são similares. A primeira permite visualizar um número que está na coluna e na linha selecionadas, a segunda permite visualizar todos os elementos já adicionados pelo usuário e suas posições.

As opções “G” e “H” permitem consultar a soma dos valores de uma linha ou de uma coluna dependendo da escolha do usuário. Sendo apenas necessário digitar uma coluna ou uma linha válida na opção correspondente.

Sendo “I” a última opção, esta permite excluir todas as informações da matriz criada, possibilitando criar uma matriz com novos parâmetros de linha e coluna.

TUTORIAL

Para compilar no Linux, deve-se abrir o terminal e, no terminal, abrir o diretório em que está localizado o código do programa `.c`. Uma vez aberto, a compilação é feita com o seguinte código: `gcc (Nome do programa).c -o (Nome do programa).out`. Com isso, será gerado um `.out` do programa. Para executá-lo, basta usar, no terminal, o código: `./(Nome do programa).out`.

Para o Windows, pode-se utilizar o auxílio de IDEs, como o CodeBlocks, para compilar o programa e executá-lo.

OUTRAS INFORMAÇÕES

As posições na matriz são dadas pela linha e coluna do número, sendo o primeiro elemento {0,0}.

O programa não irá aceitar valores que estejam fora do padrão, como opções de escolhas não informadas, linhas e colunas negativas, acrescentar um número de elementos maior que o limite da matriz, dado por: $col \times lin - n$, sendo col e lin o número de colunas e linhas, respectivamente, e 'n' o número de elementos presentes na matriz, e posições inválidas, já ocupadas por elementos ou não ocupadas, dependendo da opção escolhida.

LIMITAÇÕES

Da forma que o programa foi desenvolvido, o limite das colunas e das linhas são determinados pelo tamanho da variável escolhida, no caso o tamanho de um inteiro, o que permite a criação de matrizes extensas. Porém, ao adicionar novos elementos, a função `malloc` é chamada, que aloca memória para novas informações na memória RAM. Como não foi feito nenhum método para otimizar o uso da memória, o número máximo de elementos possíveis de serem adicionados na matriz é relativamente pequeno ao máximo de elementos possíveis de serem adicionados (os zeros não contam como elementos, a menos que o usuário deseje adicionar um).