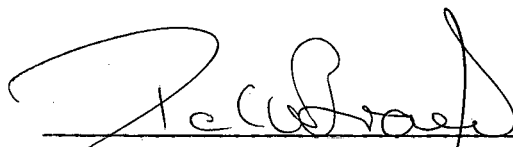


UM SISTEMA COMPUTACIONAL DIDÁTICO
PARA A SOLUÇÃO DE PROBLEMAS ENVOLVENDO
A TEORIA DOS GRAFOS

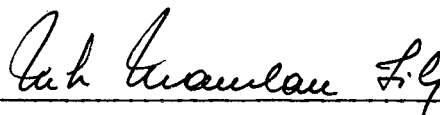
Floriano Brandão Filho

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS
DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO
DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTEN
ÇÃO DO GRAU DE MESTRE EM CIÊNCIAS (M.Sc.)


Aprovada por:



Paulo Oswaldo Boaventura Netto
(Presidente)



Nelson Maculan Filho



Ysmar Vianna e Silva Filho

RIO DE JANEIRO, RJ - BRASIL
MARÇO DE 1980

BRANDÃO FILHO, FLORIANO

Um Sistema Computacional Didático para a Solução de Problemas Envolvendo a Teoria dos Grafos (Rio de Janeiro) 1980.

VII, 276 p. 29,7cm(COPPE-UFRJ,M.Sc.,Engenharia de Sistemas, 1980).

Tese - Univ. Fed. Rio de Janeiro. Núcleo de Computação Eletrônica.

1. Teoria dos Grafos I.COPPE/UFRJ II.Título (série)

À meus pais.

S I N O P S E

O sistema proposto no presente trabalho tem o objetivo de permitir o acesso, através de um terminal de teleprocessamento, a um pacote computacional de programas relativos à Teoria dos Grafos.

Apesar de voltado primordialmente para o uso didático o sistema é de utilização bastante geral, possibilitando a resolução de variados problemas de Teoria dos Grafos.

Entre outros estão disponíveis os algoritmos para o caminho mínimo por Floyd e Dijkstra, para o fluxo máximo de Ford-Fulkerson e para a árvore parcial de valor mínimo de Kruskal.

A B S T R A C T

A computer package is presented to allow teleprocessing access to some Graph Theory algorithms.

Although designed mainly for teaching purposes, the system is of general use and may be applied to a variety of problems in Graph Theory.

The package include some classic algorithms: Ford-Fulkerson maximal flow algorithm, the Kruskal shortest spanning subtree of a graph algorithm and some shortest path algorithms (Floyd, Dijkstra).

I N D I C E

	<u>PÁGINA</u>
CAPÍTULO I - INTRODUÇÃO	
1.1 - Motivação do Trabalho	1
1.2 - Objetivos	2
1.3 - Equipamento e Linguagem	3
1.4 - Módulos de Programação	4
1.5 - Programas de Apoio e Arquivo de Grafos	6
1.6 - Notação e Nomenclatura Usada	7
CAPÍTULO II - DESCRIÇÃO GERAL	
2.1 - Diretórios	9
2.2 - Programa SISGRAFO	10
2.3 - Arquivo ARQGRA	14
2.4 - Programa INICIA	17
2.5 - Programa LEITOR	17
2.6 - Programa LISTARQ	18
2.7 - Programa APAGA	19
2.8 - Uso do Programa SISGRAFO	21
CAPÍTULO III - OPÇÕES	
3.1 - Opções de Desvio	23
3.2 - Opções de Edição	24
3.3 - Opções de Informação	28
3.4 - Opções de Alteração	30
3.5 - Opção de Armazenamento	32
3.6 - Opções de Algoritmos	33

CAPÍTULO IV - SUGESTÕES DE AMPLIAÇÃO.

4.1 - Objetivos	50
4.2 - Inclusão ou exclusão de arcos e vértices	50
4.3 - Sistema de avaliação	51
4.4 - Arquivo de informações	52
4.5 - Formatação da tela	52
4.6 - Programas de apoio	53
4.7 - Arquivo de grafos valorados	54
4.8 - Novos algoritmos	54
APENDICE A - GUIA DO USUÁRIO	56
APENDICE B - COMO USAR OS PROGRAMAS DE APOIO	165
APENDICE C - CONVERSÃO PARA TERMINAIS IBM	169
APENDICE D - TESTES EXEMPLO	172
APENDICE E - LISTAGENS DOS MÓDULOS DE PROGRAMAÇÃO	206

CAPÍTULO I - INTRODUÇÃO

1.1 - Motivação do Trabalho

A falta de programas de computador dirigidos à execução de algoritmos da Teoria dos Grupos, com finalidade didática, nos motivou grandemente na elaboração deste trabalho.

A possibilidade da utilização de um sistema de teleprocessamento permite ao aluno uma melhor racionalização no uso da máquina e a obtenção de soluções para seus problemas, muitas vezes bem simples, de um modo bastante eficiente e seguro.

A linguagem conversacional, permitida pelo teleprocessamento, possibilita ao aluno, ou mesmo a um usuário mais ambicioso, executar várias opções sobre um mesmo grafo, fornecer novos parâmetros ou até mesmo armazená-lo para uso futuro.

Além da utilidade didática, o sistema proposto permite a inserção de rotinas que poderiam processar problemas de modo operacional, eliminando-se as instruções de objetivo unicamente educacional. Desta maneira o sistema ficaria bastante versátil, ou, mesmo conforme o caso com algumas alterações no sistema atual, poderia ser construído outro que permitisse aplicações na solução de problemas práticos.

Sendo o sistema elaborado de forma modular (10), cada algoritmo ou edição de um grafo, sendo tratados como opções praticamente independentes, novos módulos, ou seja novas opções, podem ser incluídas a qualquer momento, sem maiores implicações no sistema como um todo e sem exigir o árduo trabalho de compreensão dos detalhes de todos os módulos existentes.

Desta maneira, procurou-se abrir possibilidades para ampliações futuras, que viriam justificar o presente trabalho como um ponto de partida, sem necessitar toda a elaboração inicial que permite a introdução do grafo no sistema, bastando para tanto, que seja fornecida a programação do algoritmo desejado e a ser introduzido como um novo módulo.

1.2 - Objetivos

O presente trabalho não é, nem poderia ser, exaustivo. Procurou-se, no entanto, abranger alguns tópicos principais da teoria, desenvolvendo os algoritmos que pudessem vir a ser de maior interesse no transcorrer de um curso introdutório sobre a Teoria do Grafos (3).

O sistema apresentado não tem o objetivo de permitir ao aluno um aprendizado da teoria pela sua simples utilização; no entanto possibilita um melhor acompanhamento do curso pela análise dos relatórios apresentados.

A cada execução do sistema são imputadas penalidades por eventuais erros de conceituação cometidos, o que permite uma avaliação do grau de desenvolvimento adquirido no curso.

Tendo em vista o objetivo didático, foram introduzidas muitas opções de informação sobre o grafo, as quais não teriam grande utilidade ao se resolver um problema prático específico. Assim, sempre que possível, é permitido ao usuário um salto para evitar a saída destas informações que, no caso, seriam irrelevantes.

Levou-se em consideração na escolha dos algoritmos a serem implantados, não somente a finalidade didática do trabalho, mas também a sua possível aplicação a problemas reais, como estudos de flu

xos, a existência de circuitos, ciclos, caminhos etc.:

Objetivando a comparação entre algoritmos que possibilitam obter-se informações semelhantes, algumas opções são repetidas, porém executando rotinas diferentes, como é o caso dos algoritmos de caminho mínimo, foram incluídos o de Floyd e o de Dijkstra.

Com a criação de um arquivo de dados, pretende-se implantar grafos que possuam peculiaridades de interesse e que possam ilustrar alguns pontos da teoria; neste arquivo poderão também ser colocados futuramente grafos de um uso mais específico.

1.3 - Equipamento e linguagem.

O sistema foi todo preparado para operar em terminais de teleprocessamento do computador Burroughs B6700, instalados no Campus da Universidade Federal do Rio de Janeiro, (vide configuração na referência 8).

Com algumas pequenas alterações básicas o sistema poderá operar em qualquer computador Burroughs que possa trabalhar com o CANDE (Command And Edit) (7), para terminais de teleprocessamento. A conversão para outro equipamento irá fatalmente envolver maiores mudanças, que dependerão do compilador disponível; algumas sugestões neste sentido serão apresentadas no Apêndice C.

Apesar das desvantagens aparentes na escolha do FORTRAN (Formula Translation) como linguagem básica para a programação dos módulos do sistema, duas razões bastante fortes nos levaram a sua escolha:

- (1) Dado nossa experiência, pelo uso continuado desta linguagem por alguns anos, o que veio a facilitar sobremaneira a elaboração dos programas, e
- (2) Visto ser o FORTRAN, praticamente, uma linguagem comum a quase todos os equipamentos, o que possibilitaria a adaptação dos módulos de programação com maior facilidade.

1.4 - Módulos de Programação

Constituem os módulos de programação, todas as rotinas auxiliares do sistema e mais as opções que permitem a escolha do trabalho a ser executado.

As rotinas auxiliares foram implantadas com o objetivo de agrupar certos passos, que seriam repetitivos em boa parte das opções.

Como rotinas auxiliares estão incluídas, entre outras: rotina que permite uma parada da tela para observação, rotinas de mensagens enviadas, rotinas para teste de valores válidos, rotinas para a inserção ou retirada de elementos do arquivo de grafos e rotinas de preparação da tela para entrada de dados.

Por não ser este o objetivo principal do presente trabalho, não será apresentada uma descrição detalhada das rotinas auxiliares, porém uma listagem de todas elas pode ser encontrada no Apêndice E.

As opções são módulos de programação independentes, que permitem a execução de uma rotina qualquer referente à Teoria dos Grafos.

Eventualmente uma opção pode automaticamente, chamar outra retornando em seguida à opção anterior. Isto ocorre, por exemplo, quando se escolhe um algoritmo que só é válido para grafos valorados e o grafo ainda não foi valorado, ou um algoritmo para grafos não orientados e o grafo está orientado; em ambos os casos será fornecida uma mensagem de advertência.

Algumas opções tem a finalidade única de fornecer informações sobre o grafo que está sendo usado. Estas opções podem ser chamadas a qualquer momento, desde que o grafo já tenha sido editado. Como exemplo temos a opção que mostra na tela os sucessores de cada vértice, a opção que coloca na tela o semigrau exterior dos vértices, a opção que mostra na tela a matriz de adjacência e a opção para imprimir a matriz de adjacência.

Devido a impossibilidade de se apresentar uma lista com todas as opções na tela de uma só vez, isto é feito usando-se três listas com dez opções cada uma, com a última opção de cada lista sendo um ponteiro para a próxima lista.

As opções de algoritmo foram bastante orientadas para a finalidade didática do trabalho; assim, sempre que possível, foram implantadas mensagens de informações ou advertência. Algumas destas mensagens podem ser evitadas durante a execução, quando eventualmente já se possui a informação ou não se deseja saber; em alguns casos, porém, as mensagens são realmente impressas, independentemente do controle do usuário.

Uma descrição mais detalhada de todas as opções é feita no Capítulo III.

A limitação imposta pelo sistema proposto permite somente operar com grafos de no máximo 40 vértices.

1.5 - Programas de Apoio e Arquivo de Grafos.

Para se trabalhar com o sistema de uma maneira mais rápida, principalmente no que se refere a entrada de dados, foi criado um arquivo onde pode ser colocado um certo número de grafos para uso futuro.

Para que este arquivo pudesse ser periodicamente atualizado, foram elaborados alguns programas ditos de apoio, que permitem o acesso a este arquivo para vários tipos de atualização, tais como a inclusão de novos grafos, listagem dos grafos existentes ou a exclusão de grafos e a compactação do arquivo.

Estes programas não fazem parte do sistema propriamente dito, ou seja não pertencem aos módulos de programação, porém constituem parte importante do acervo total para permitir, a qualquer tempo, obter-se informações seguras a respeito dos dados armazenados.

Informações de como e quando se deve usar estes programas podem ser encontradas no Apêndice B.

1.6 - Notação e Nomenclatura usada.

É sabida a grande dificuldade que existe para se estabelecer uma nomenclatura e uma notação geral para a Teoria dos Grafos. Existem grandes diferenças de um autor para outro; desta maneira é quase impossível seguir-se apenas um deles. De um modo geral, a notação aqui usada segue a de Berge (1,2).

Estão definidos abaixo alguns dos conceitos usados:

- 1) $G = (X , U)$: definição de um grafo G
- 2) $X = \{x_1 , x_2 , \dots , x_n\}$: vértices ou nós do grafo
- 3) $n = | X |$: número de vértices ou ordem da estrutura
- 4) $U = \{u_j \mid j \in I\}$: para um 1 - grafo não orientado corresponde o conjunto de partes de X 2 a 2 e são ditas arestas. Para um 1 - grafo orientado corresponde um subconjunto do produto cartesiano $X \times X$ e são ditos arcos.
- 5) (x_i , x_j) : definição de um arco do vértice x_i para x_j
- 6) $m = | U |$: número de arcos/arestas do grafo.
- 7) $A = [a_{ij}]$: matriz de adjacência (booleana)
 $a_{ij} = 1 \iff \exists (x_i , x_j)$
 $a_{ij} = 0 \iff \nexists (x_i , x_j)$
- 8) $\mu (x_i , x_j)$: Caminho entre x_i e x_j

9) $D = [d_{ij}]$: matriz de distâncias

$$d_{ij} = 0 \iff i = j$$

$$d_{ij} = \infty \iff \nexists \mu(x_i, x_j) \quad x_i \neq x_j$$

$$d_{ij} < \infty \iff \exists \mu(x_i, x_j) \quad x_i \neq x_j$$

10) $V = [v_{ij}]$: matriz de valores ou custos

11) $R = [r_{ij}]$: matriz de roteamento

12) $S_o(x_i)$: afastamento exterior do vértice x_i

$S_t(x_i)$: afastamento interior do vértice x_i

13) $\rho_o(G)$: raio exterior de G

$\rho_t(G)$: raio interior de G

14) $\delta_o(G)$: diâmetro exterior de G

$\delta_t(G)$: diâmetro interior de G

15) $d^+(x_i) = \sum_{j=1}^n a_{ij}$: semigrau exterior de x_i

16) $d^-(x_i) = \sum_{i=1}^n a_{ij}$: semigrau interior de x_i

CAPÍTULO II - DESCRIÇÃO GERAL

2.1 - Diretórios

Foram montados dois diretórios, que contêm os programas ou arquivo de dados, de acordo com a sua finalidade.

Os diretórios estão assim constituídos:

SISTEMA	{	SISGRAFO
		ARQGRA

SUPORTE	{	INICIA
		LEITOR
		LISTARQ
		APAGA

O diretório SISTEMA possui o programa, denominado SISGRAFO, que contém todos os módulos de programação. Este programa poderá ser chamado por qualquer usuário, que possua autorização para uso dos terminais de teleprocessamento do B6700.

No arquivo ARQGRA estão armazenados alguns grafos de interesse acadêmico. O acesso a este arquivo pode ser feito tanto pelos programas de apoio como pelo SISGRAFO. Qualquer usuário pode utilizar este arquivo apenas para leitura (via SISGRAFO), porém se desejar gravar algum grafo de interesse deverá conhecer uma senha autorizada, que será solicitada pelo SISGRAFO quando for executada a opção de gravação ou ainda usar o programa de apoio LEITOR pertencente ao diretório SUPORTE.

Os programas do diretório SUPORTE permitem a manipulação e gerenciamento dos grafos armazenados no arquivo ARQGRA. Os programas deste diretório devem ser processados com bastante cautela e portanto não serão de uso corrente por qualquer usuário; no entanto a execução de alguns destes programas deve ser feita periodicamente para que se possa otimizar a utilização do espaço disponível para o arquivo ARQGRA.

2.2 - Programa SISGRAFO

Neste programa, que faz parte do diretório SISTEMA, estão embutidos todos os módulos de programação, constituídos pelas rotinas auxiliares e todas as opções.

A sua execução é conduzida de forma conversacional entre o usuário e o sistema, permitindo uma intervenção contínua para executar ou não determinadas partes do programa.

Existem ao todo 31 opções, numeradas de 0 a 30, cada uma delas tendo uma função específica. De acordo com o seu objetivo as opções podem ser classificadas em seis grupos diferentes:

1 - Opções de Desvio.

Permitem o desvio para o encerramento da execução do programa ou a apresentação de uma das listas para escolha da opção a ser executada.

2 - Opções de Edição.

A princípio pode-se escolher qualquer opção, porém se o grafo a ser usado ainda não foi editado, esta escolha só pode recair sobre uma das opções de edição, caso contrário será dada uma mensagem de advertência para a escolha da opção adequada.

Uma exceção é feita para o caso de optar-se pelo encerramento de execução do programa.

As opções de edição permitem a escolha do grafo a ser usado de quatro maneiras diferentes. Pode-se gerar aleatoriamente um grafo, escolher um grafo armazenado no arquivo, ler um grafo fornecido pelo terminal ou ainda fornecer parte do grafo pelo terminal e gerar mais arcos aleatoriamente.

É importante notar que o sistema só trabalha com um grafo de cada vez, assim ao ser chamada uma opção de edição o grafo anterior, se existir, será destruído, a menos que tenha sido gravado anteriormente.

3 - Opções de Informação

Estas opções quando solicitadas fornecem as informações preliminares sobre o grafo em uso, tais como o número de sucessores e quais são eles para cada vértice, além de permitirem a observação da matriz de adjacência na tela ou a sua impressão.

As informações constantes destas opções já se encontram disponíveis, uma vez que o grafo tenha sido editado.

4 - Opções de Alteração

Estas opções permitem a mudança da estrutura inicial do grafo em uso, pode-se, por exemplo transformar um grafo orientado em um grafo não orientado, ou ainda fornecer valores para os arcos de um grafo.

5 - Opção de Armazenamento

Esta opção permite, mediante o conhecimento de uma senha autorizada, o acesso ao arquivo ARQGRA para a gravação de um grafo, que se julgue de interesse para uso posterior.

6 - Opções de Algoritmos

Estas opções permitem o uso dos algoritmos implantados que serão aplicados ao grafo presente no sistema naquele momento (ou melhor, ao último grafo que foi editado).

Alguns destes algoritmos solicitam maiores informações. A grande maioria dos resultados é apresentada na tela, porém em alguns casos deve-se responder se se deseja ou não um resultado impresso.

É importante notar que, uma vez executada uma das opções de edição, qualquer das outras pode ser executada na ordem que se desejar. Porém, tendo em vista a função didática, deve-se estar atento para algoritmos que sã são válidos, por exemplo, para grafos orientados (ou não orientados), ou ainda válidos somente para grafos já valorados; nestes casos a ordem de chamada das opções é levada em conta, incrementando-se um erro ao contador sempre que esta sequência não seja obedecida.

O programa SISGRAFO está constituído de três partes fundamentais:

PARTE 1: Contém as mensagens de inicialização, como apresentação do sistema, solicitação do nome e número do usuário e se deseja ou não um relatório final impresso. Nesta parte estão ainda as mensagens de finalização, como o dia e hora do início e término do trabalho e o resultado do con-

tador de erros.

PARTE 2 : Constituída por uma única rotina cuja finalidade é servir de chaveamento para chamada de todas as opções solicitadas.

PARTE 3 : Onde estão colocadas todas as opções propriamente ditas.

Um fluxo geral do programa é apresentado na Figura II.1.

Uma vez solicitado um relatório final impresso, o sistema emitirá uma listagem contendo todos os passos executados no terminal, intercalados com as eventuais solicitações de impressão.

O Apêndice A corresponde a um Guia do Usuário, onde estão ilustradas todas as opções e como elas devem ser operadas, para que o programa SISGRAFO possa ser executado de uma maneira segura e eficiente.

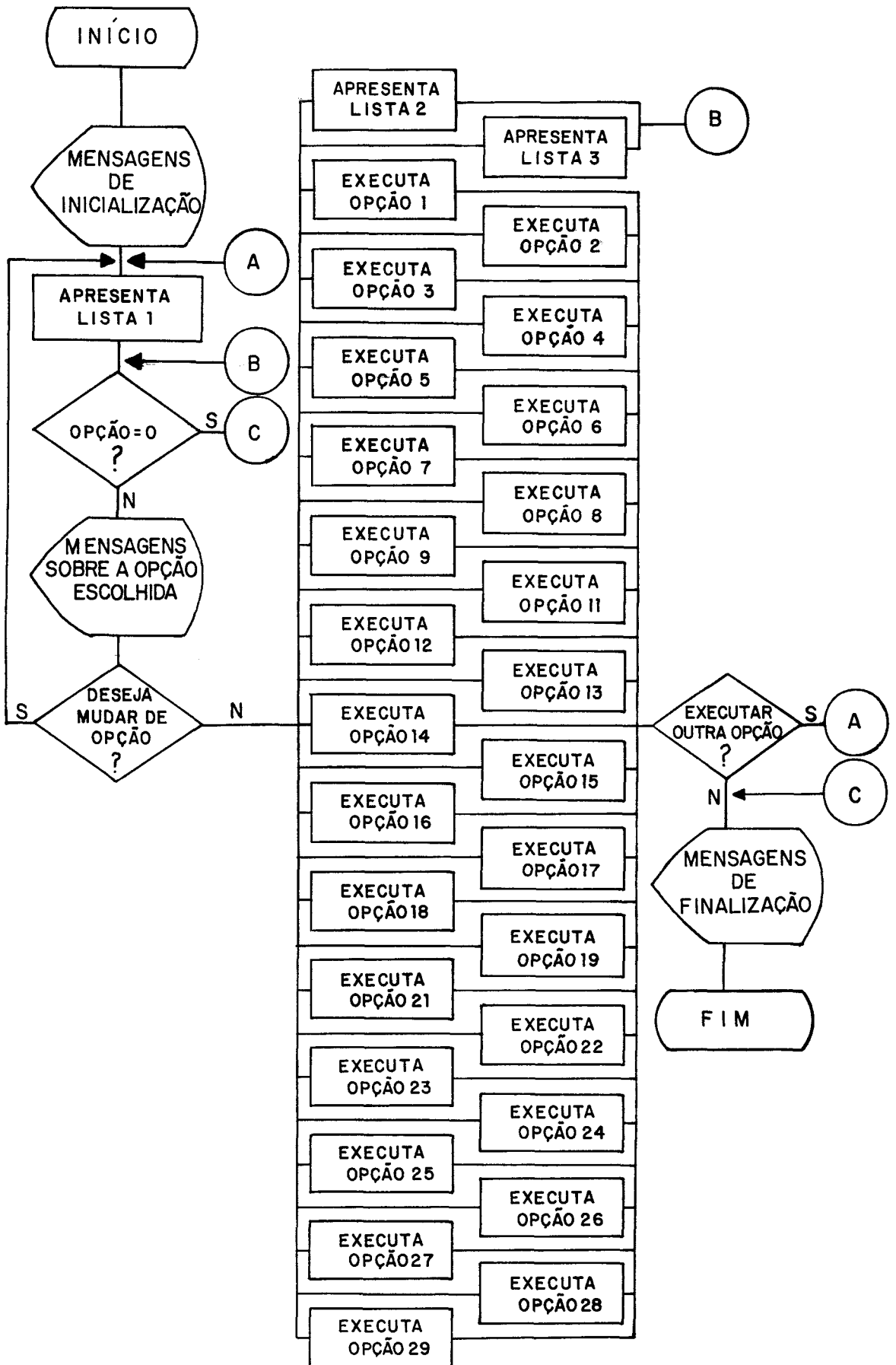


Figura II - 1

2.3 - Arquivo ARQGRA

Este arquivo foi elaborado visando a maior economia de espaço possível; assim sendo, s̄o foram armazenados os dados estritamente necess̄arios à definiç̄ão do grafo.

O arquivo ARQGRA pertence ao diretório SISTEMA.

Pelo fato de se usar um armazenamento estruturado, alḗm dos dados referentes aos grafos foi necess̄ario introduzir-se alguns ponteiros.

Para que se possa otimizar o espaço usado pelo arquivo, limitou-se o n̄mero de registros que podem ser colocados, porḗm este limite pode ser alterado se for desej̄avel.

Os quatro primeiros registros contḗm informaç̄ões sobre o arquivo e dados de controle para inicializaç̄ão do acesso:

Registro 1: ũltima posiç̄ão gravada (TOP)

Registro 2: n̄mero do ũltimo grafo armazenado (LAST)

Registro 3: - 1 (simula o n̄mero de um grafo inexistente)

Registro 4: posiç̄ão do primeiro grafo armazenado

Para cada grafo armazenado existem os seguintes registros:

Registro 1: n̄mero do grafo (NG)

Registro 2: posiç̄ão de inīcio do pr̄ximo grafo armazenado (NEXT)

Somente s̄o armazenados para cada grafo os v̄rtices que possuem o n̄mero de sucessores maior ou igual a 1. Desta maneira, para cada v̄rtice dentro de cada grafo existem os seguintes registros:

Registro 1 : Número do vértice (VERT)

Registro 2 : Número de sucessores do vértice (NSUC)

A seguir existem tantos registros quantos forem os sucessores do vértice em questão, contendo cada um o número de um sucessor.

Como ilustração a Figura II.2 apresenta um exemplo do arquivo, em uma situação correspondente ao armazenamento de quatro grafos:

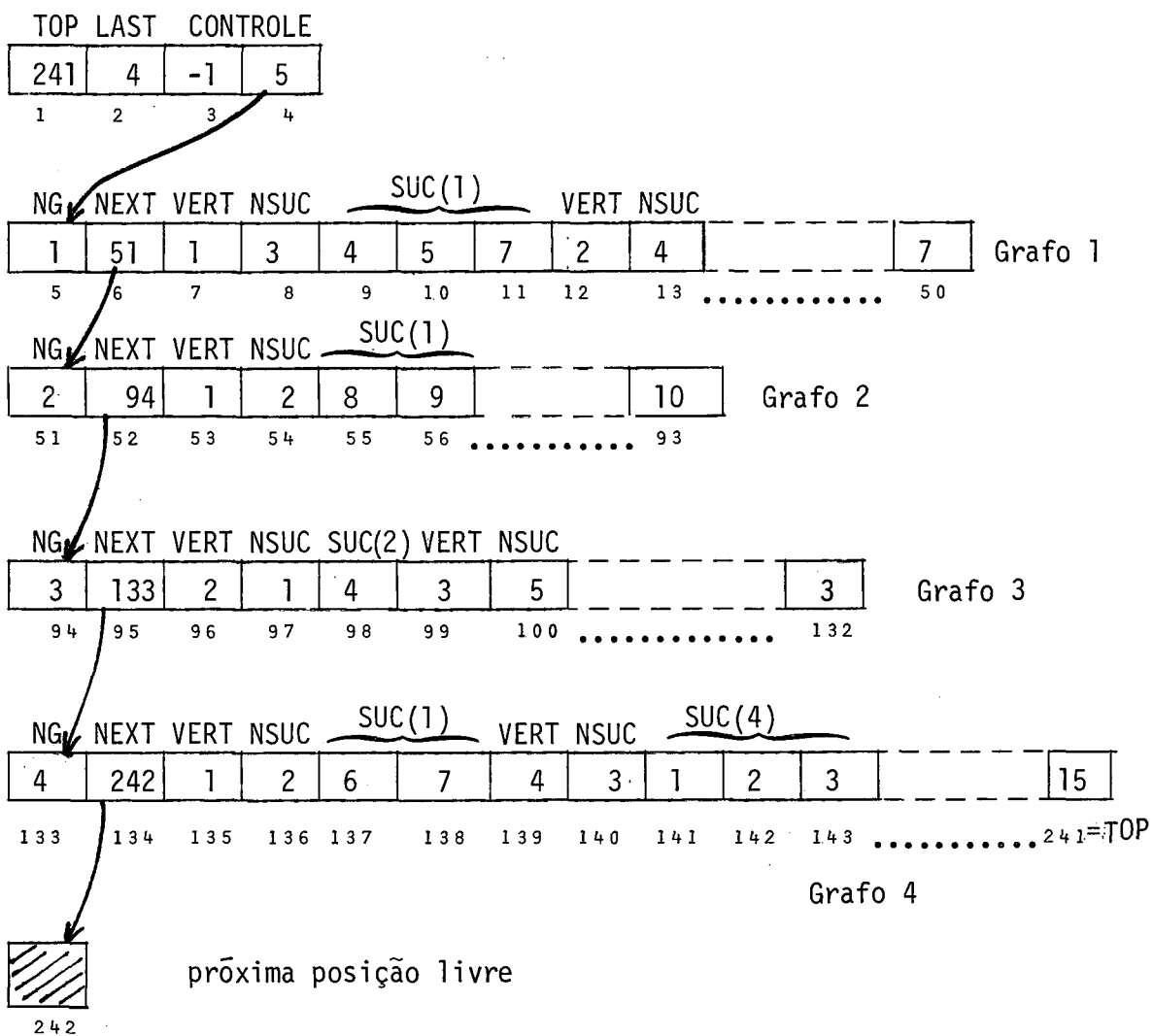


Figura II.2

O número de registros que serão ocupados para armazenar cada grafo pode ser dado por:

$$NR = NARC + 2 NVER + 2$$

Onde:

NARC - Número de arcos do grafo

NVER - Número de vértices que possuem pelo menos um sucessor.

O número 2 adicional corresponde a dois registros, um para o número do grafo e outro para indicar a posição do próximo grafo.

O número total de registros para armazenar N grafos será dado por:

$$NRT = 2 N + 4 + \sum_{i=1}^N (NARC_i + 2 NVER_i)$$

O número 4 adicional corresponde aos registros iniciais de informação sobre o arquivo e os registros de controle.

A manipulação deste arquivo é feita pelos programas de apoio, pertencentes ao diretório SUPORTE, que serão descritos a seguir.

Todos os grafos são armazenados como se fossem grafos orientados. Assim para se armazenar os grafos não orientados é vantajoso fornecer para o armazenamento, somente os arcos pertencentes ao triângulo inferior da matriz de adjacência. Quando se desejar usar este grafo pode-se então chamar a opção de alteração que desorienta o grafo.

2.4 - Programa INICIA

Este programa, pertencente ao diretório SUPORTE, tem a finalidade de inicializar o arquivo de grafos ARQGRA.

Quando executado ele zera os cinco primeiros registros do arquivo, indicando assim, para qualquer programa que tente utilizar o ARQGRA, que ele está vazio.

O INICIA é usado somente em dois casos especiais:

- (1) Quando se deseja gravar o arquivo pela primeira vez, e
- (2) Quando se deseja apagar o arquivo, isto é, eliminar todos os grafos armazenados de uma só vez.

Quando se desejar apagar apenas alguns grafos, deve-se usar o programa APAGA, descrito mais adiante no item 2.7 deste capítulo.

Terminada a execução, será fornecida uma mensagem, informando que o arquivo foi inicializado e que não existe nenhum grafo armazenado.

2.5 - Programa LEITOR

Este programa, pertencente ao diretório SUPORTE, tem a finalidade de ler um ou mais grafos via cartão e armazená-los no arquivo ARQGRA.

Sempre que possível, deve-se usar este programa, em preferência à opção de armazenamento existente no programa SISGRAFO, pois o uso desta opção implica, muitas vezes, em uma leitura do grafo pelo terminal, o que se torna muito lento, principalmente quando se trata de grafos contendo muitos arcos, que devem ser digitados

pelo console.

Quando houver dados incoerentes para algum v̄ertice do grafo que esta sendo armazenado, ser̄a dada uma mensagem de advertência, o grafo em quest̄ão n̄o ser̄a armazenado e o programa passar̄a a armazenar o pr̄oximo grafo.

A medida em que os grafos v̄o sendo gravados, eles recebem um n̄mero sequencial, a partir do ũltimo grafo que estava armazenado.

Como mensagem final, ẽ fornecido o n̄mero que cada grafo armazenado recebeu, bem como uma listagem dos sucessores de cada v̄ertice.

2.6 - Programa LISTARQ.

Este programa, pertencente ao diret̄orio SUPORTE, tem a finalidade de imprimir todos os grafos que est̄o gravados em ARQGRA.

Para cada grafo armazenado s̄o fornecidos os seguintes dados:

- (a) N̄mero do grafo, que deve ser usado sempre que se desejar referenciar a ele;
- (b) N̄mero de v̄ertices;
- (c) N̄mero de arcos;
- (d) Uma lista dos sucessores de cada v̄ertice.

Ao final, são fornecidas mensagens sobre o arquivo tais como:

- (a) O número de grafos armazenados,
- (b) O número do último grafo armazenado, e
- (c) Valor da última posição gravada, acompanhado do valor máximo permitido, para possibilitar uma avaliação do espaço ainda existente.

2.7 - Programa APAGA

Este programa, pertencente ao diretório SUPORTE, tem a finalidade de excluir um ou mais grafos do arquivo ARQGRA.

Uma vez indicado qual o grafo que se deseja eliminar, o programa atualiza todos os ponteiros referentes a este grafo e depois copia o restante do arquivo ocupando as posições deixadas livres por ele.

Desta maneira, a cada grafo apagado, o arquivo é compactado, liberando espaço para o armazenamento de outros grafos de interesse.

Para cada grafo eliminado serão dadas as seguintes informações:

- (a) O número do grafo apagado,
- (b) O número do último grafo armazenado, e
- (c) Valor da última posição gravada.

Usando o exemplo da Figura II.2 e apagando-se o grafo número 2, o arquivo ficaria com a configuração apresentada pela Figura II.3, economizando assim 43 posições no novo armazenamento.

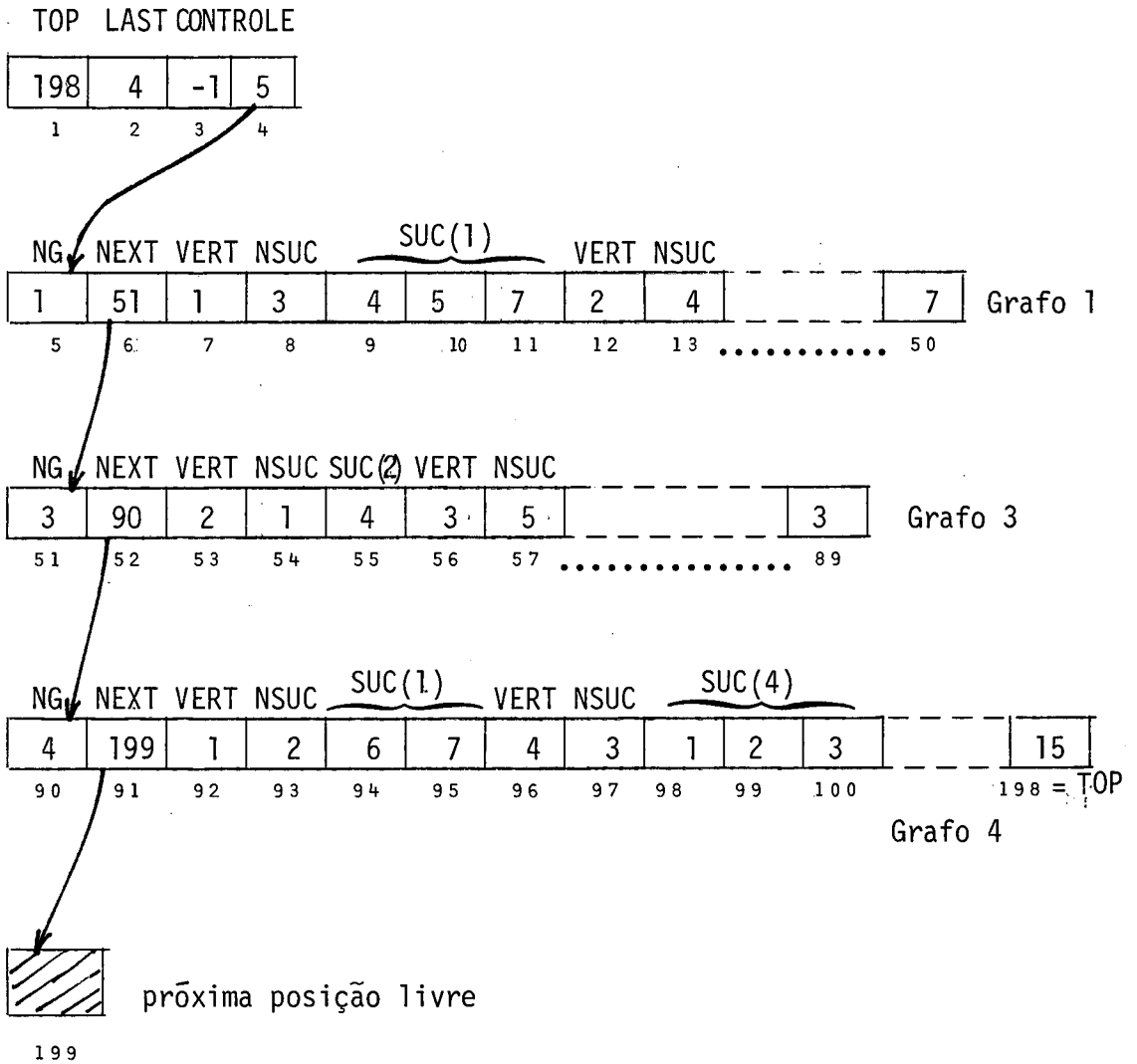


Figura II.3

2.8 - Uso do Programa SISGRAFO 1/.

Existem duas maneiras pelas quais se pode operar com o programa SISGRAFO:

(1) Executar o programa pelo terminal.

Neste caso estará alocando uma área para sua execução chamada task 2/. Dependendo da capacidade definida pelo MCP (Master Control Program) (8), o número de tasks, liberadas pelo sistema operacional, para uso em terminal, pode ser bastante restrito.

Assim durante todo o tempo de execução do programa uma task estará sendo retida, restringindo ainda mais a disponibilidade para outros usuários.

A vantagem da execução do SISGRAFO por este processo é que o tempo de espera para o início da execução é bastante curto, de um modo geral, quando comparado com o tempo de um job colocado na fila de espera.

Outra vantagem reside no fato de não se necessitar estabelecer limites para os tempos de processamento e de entrada e saída ou para o número de linhas impressas.

1/ Levou-se em consideração na elaboração deste item o uso de um sistema Burroughs B6700.

2/ Uma task é todo e qualquer processo iniciado por um job (8).

(2) Colocar o programa numa fila.

Neste caso o programa é colocado em uma fila para seleção, independente do terminal. Desta maneira não será usada nenhuma das tasks reservadas para os terminais e assim o terminal que se esta usando passa a funcionar apenas como um meio de entrada e saída.

Via de regra o programa levará um pouco mais de tempo para iniciar o processamento, e neste caso deve-se estar atento para o início da execução, pois o sistema operacional enviará mensagens, para o terminal determinado no job, as quais devem ser respondidas pelo usuário.

Quanto a execução do programa em si não existe nenhuma diferença entre um tipo e outro, porém desde que o sistema operacional apresente boa flexibilidade para o uso de terminais pode-se notar uma larga vantagem na execução do programa pelo terminal 3/.

3/ Este não é o caso do sistema instalado no campus da UFRJ, uma vez que o número de tasks liberadas para uso em terminal é ainda bastante limitado. Existem ainda as classes transparentes; e os jobs executados pelo terminal tem também os tempos de processamento, entrada e saída e o número de linhas limitadas.

CAPÍTULO III - OPÇÕES

3.1 - Opções de Desvio

As opções de desvio permitem ao usuário escolher qual a sua decisão para continuação dos trabalhos.

Assim existe uma opção que, uma vez escolhida, desvia a execução para as mensagens de finalização e não permitem mais o retorno ao programa, a menos que se inicie a execução novamente.

Ao final da execução de qualquer opção, pergunta-se sempre se o usuário deseja ou não executar uma nova opção, uma resposta negativa desviará o programa para as mensagens de finalização. Deste modo existem duas maneiras de se encerrar os trabalhos: uma pela opção de desvio para o final e outra pela resposta negativa ao fim de cada opção.

A resposta positiva à pergunta acima levará à apresentação na tela da primeira lista de opções. A última opção desta lista apresentará a segunda, a última opção da segunda apresentará a terceira, e a última desta retornará o programa para apresentação da primeira lista. Isto se deve à impossibilidade de se apresentar na tela, de maneira legível, todas as opções de uma só vez.

Sabendo-se previamente o número da opção que se deseja executar, ela pode ser chamada a qualquer tempo, mesmo que não esteja presente na lista que estiver apresentada na tela naquele instante, porém a primeira lista será sempre apresentada

Uma vez escolhida a opção desejada o programa passará a executar a rotina específica e todo o controle passará então a ser desta rotina.

3.2 - Opções de Edição

Estas opções constituem o ponto de partida para execução de qualquer trabalho utilizando o programa SISGRAFO.

A finalidade destas opções é definir para o sistema qual será o grafo usado, bem como ele será obtido. Portanto sempre que se iniciar a execução de SISGRAFO deve-se obrigatoriamente chamar uma das opções de edição, caso contrário uma mensagem alertará para o fato.

Os quatro modos de se editar um grafo, que pode ser usado pelas opções implantadas, estão descritos nos itens a seguir.

3.2.1. - Geração de um Grafo Aleatoriamente.

Sendo o grafo gerado definido pelo par ordenado $G = (X,U)$ onde X representa o conjunto dos vértices e U os arcos do grafo, o algoritmo criado irá gerar os arcos $u \in U$ segundo uma probabilidade p .

O algoritmo não gera laços, ou seja não existirão os arcos (x_i, x_i) .

Pode-se limitar o número de arcos a serem gerados, e neste caso a probabilidade p da existência de arcos estará definida por este número. Sendo m o número de arcos a serem gerados, o algoritmo considera, no caso de $m=0$, que os arcos devem ser gerados segundo uma probabilidade p fornecida.

Para os grafos não orientados, serão gerados os arcos nos dois sentidos, ou seja, se existir (x_i, x_j) , existirá também (x_j, x_i) .

Os dados necessários para o algoritmo são: a orientação do grafo, uma semente para geração dos valores aleatórios, o número de vértices(n), o número de arcos(m), e se $m=0$ a probabilidade da existência de um arco(p).

Algoritmo:

1. Fornecer: Orientação, semente aleatória, n e m .
2. Se $m > 0$ prosseguir; caso contrário, ir para o passo 3.

Fazer $p = \frac{m}{n(n-1)}$ para grafo orientado.

Fazer $p = \frac{2m}{n(n-1)}$ para grafo não orientado.

Fazer $K = 0$

Ir para passo 4.

3. Fornecer p .
4. Fazer $i=1$.
5. Fazer $j=1$.
6. Se $i = j$ ir para o passo 10.
7. Gerar um número aleatório r . ($0 < r \leq 1$)
8. Se $r > p$ ir para o passo 10.
9. Gerar o arco (x_i, x_j)
Se grafo não orientado gerar o arco (x_j, x_i)
Fazer $K = K + 1$
10. Se $m \neq 0$ e $K \geq m$: encerra a execução e foram gerados m arcos
11. Se $j < n$ fazer $j = j + 1$, ir para o passo 6
12. Se $i < n$ fazer $i = i + 1$, ir para o passo 5

Para os grafos não orientados o algoritmo fica simplificado devido a simetria, e neste caso o índice j deve variar desde $i + 1$ até n .

Se $m > 0$ o valor fornecido é criticado em função do máximo número de arcos que o grafo em questão pode ter:

$$\text{Se grafo orientado} \quad m \leq n (n - 1)$$

$$\text{Se grafo não orientado} \quad m \leq \frac{n (n - 1)}{2}$$

3.2.2 - Uso de um Grafo Armazenado

Neste caso o grafo a ser usado já foi armazenado previamente, no arquivo ARQGRA descrito no item 2.3 do Capítulo II.

Para se usar esta opção basta apenas fornecer o número do grafo desejado.

É importante lembrar que todos os grafos armazenados são orientados. Pode-se eliminar a orientação usando a opção própria para tal, que será vista mais adiante.

O sistema de crítica dos dados, nesta rotina verifica somente a existência ou não do grafo no arquivo.

3.2.3 - Leitura de um Grafo pelo Terminal

Este modo de edição não é muito recomendado para grafos que possuem muitos arcos, pois o tempo de entrada dos dados pelo terminal pode se tornar muito grande. Neste caso seria uma melhor providência que o grafo fosse armazenado previamente, utilizando o programa de apoio LEITOR, que permite armazenar o grafo proveniente de dados em cartão.

Uma vez fornecido o número de vértices do grafo, será então solicitado para cada vértice o seu semigrau exterior, e após a tela ficar formatada, deve-se fornecer os sucessores do vértice em questão.

A entrada do valor zero para o semigrau exterior de um dado vértice indica que ele não possui sucessores.

A rotina possui um sistema de crítica para os dados fornecidos, não permitindo a entrada de um vértice repetido como sucessor de outro, ou o número de um sucessor maior que o número de vértices do grafo, ou o semigrau exterior maior que o número de vértices, ou ainda valores menores ou iguais a zero (exceção feita ao semigrau exterior).

Se o grafo for não orientado, pode-se fornecer apenas os sucessores que se encontram no triângulo inferior da matriz de adjacência, pois neste caso os dados para a matriz armazenada serão simetrizados em relação à diagonal principal.

3.2.4 - Lê um grafo pelo terminal e gera mais arcos.

Esta rotina consiste em uma combinação dos modos apresentados nos itens 3.2.1 e 3.2.3.

Torna-se bastante útil, quando se deseja que um grafo tenha determinada, propriedade, por exemplo exista um caminho entre dois vértices quaisquer, porém uma vez garantida tal propriedade o número de arcos existentes será irrelevante.

A rotina para leitura será chamada, e neste caso deve-se fornecer as informações solicitadas para todos os vértices, incluindo também aqueles para os quais o semigrau exterior e interior sejam zero.

Quando for chamada a rotina para geração aleatória será solicitado o valor da semente e a probabilidade da existência de um arco. Não será solicitado no entanto o número de arcos que se deseja colocar. A rotina implantada segue praticamente o algoritmo apresentado no item 3.2.1.

As críticas existentes são as mesmas já apresentadas anteriormente nas rotinas envolvidas.

3.3 - Opções de Informação.

Estas opções foram implantadas com o objetivo de permitir ao usuário ter algumas informações básicas sobre o grafo que está sendo usado, independente dos dados fornecidos quando o grafo é editado.

Estas informações podem muitas vezes ser bastante úteis para se analisar o grafo antes de chamar algum algoritmo específico.

3.3.1 - Sucessores dos vértices

Quando solicitada esta opção apresentará na tela, para cada vértice, o seu semigrau exterior e quais são os seus sucessores.

Os vertices serao apresentados na tela em grupos de tres de cada vez, havendo neste instante uma parada da tela para observaao.

3.3.2 - Semigrau Exterior

Esta opao apresenta na tela uma lista de todos os vertices do grafo em uso, com o seu respectivo semigrau exterior.

3.3.3 - Imprime a Matriz de Adjacencia

Uma vez solicitada esta opao emitira uma listagem contendo a matriz de adjacencia na forma booleana, para o grafo em uso.

Esta opao admite qualquer numero de vertices aceito pelo sistema, o que nao ocorre com a opao seguinte.

3.3.4 - Mostra na Tela a Matriz de Adjacencia

Quando chamada colocara na tela a matriz de adjacencia, na forma booleana, do grafo em uso.

Em vista de problemas de legibilidade, esta opao so e permitida para grafos que possuam no maximo 15 vertices; caso se deseje a informaao sobre a matriz para grafos maiores, deve-se recorrer a opao do item anterior.

3.4 - Opções de Alteração

Estas opções tem a finalidade de alterar a estrutura do grafo que foi previamente editado. As alterações efetuadas sã̃o existem durante o tempo de execuçã̃o do SISGRAFO, nã̃o implicando em nenhuma mudançã̃a nos grafos armazenados no arquivo.

3.4.1 - Dar Valores aos Arcos

Esta rotina permite que sejam fornecidos valores para todos os arcos existentes no grafo em uso.

Pode-se optar por fornecer um valor constante para todos os arcos, ou entã̃o usar valores diferentes.

Para os ṽertices que nã̃o estiverem ligados serã̃o imputado um valor considerado infinito, o qual levarã̃o a impressã̃o de asteriscos no campo de saı̃da.

Sendo $A = [a_{ij}]$ a matriz de adjacência e $V = [v_{ij}]$ a matriz de valores, a rotina segue a seguinte norma:

Se $a_{ij} = 1 \Rightarrow$ solicitarã̃o a informaçã̃o de v_{ij}

Se $a_{ij} = 0 \Rightarrow v_{ij} = \infty$

Sendo optado um valor constante para os ṽertices:

Se $a_{ij} = 1 \Rightarrow v_{ij} = \text{constante}$

Se $a_{ij} = 0 \Rightarrow v_{ij} = \infty$

Os valores de i e j variam de 1 até o número de vértices do grafo.

Para os grafos não orientados, serão solicitados apenas os valores para os arcos do triângulo inferior da matriz de adjacência, e os valores restantes serão simetrizados tendo como base os valores fornecidos.

Uma vez encerrada a execução desta rotina pode-se chamar qualquer opção que exija um grafo valorado. Se for desejado, a rotina pode ser chamada outras vezes para fornecer uma nova valoração, o que implica sempre na substituição dos valores anteriormente fornecidos.

Como já foi dito anteriormente, sempre que for chamada uma opção de edição, o grafo anterior, se existir, será destruído. Assim o novo grafo editado inicializa com um indicador de que este último grafo ainda não foi valorado.

3.4.2 - Eliminar a Orientação de um Grafo

Esta rotina permite eliminar a orientação de um grafo orientado, ou seja indicar a existência de arcos nos dois sentidos.

Para os grafos valorados, este fato também é levado em consideração.

A rotina obedece as seguintes normas:

$$\text{Se } a_{ij} = a_{ji} = 1 \Rightarrow v_{ji} = v_{ij} = \max(v_{ij}, v_{ji})$$

$$\left. \begin{array}{l} \text{Se } a_{ij} = 1 \\ \text{e} \\ a_{ji} = 0 \end{array} \right\} \Rightarrow v_{ji} = v_{ij} \quad \text{e} \quad a_{ji} = a_{ij} = 1$$

Onde i e j variam de 1 até o número de vértices.

Uma vez encerrada a execução desta rotina, o grafo anterior passa a não mais existir para a presente execução do SISGRAFO.

3.5 - Opção de Armazenamento

Esta opção permite colocar no arquivo ARQGRA um grafo que tenha sido anteriormente editado.

O grafo, ao ser armazenado, segue o esquema que foi apresentado no ítem 2.3 do Capítulo II.

Para se usar esta rotina é necessário, na hora de execução, fornecer uma senha autorizada para o acesso, que permite a gravação no arquivo. O usuário terá três chances para fornecer a senha correta.

Caso seja excedida a capacidade de gravação no arquivo, uma mensagem de advertência será enviada, e o grafo em questão não terá nenhum de seus valores armazenados.

Se a gravação foi terminada com êxito serão dadas as indicações para futuros acessos ao arquivo, e também o número que o grafo armazenado recebeu. Este é o número que deve ser usado sempre que for desejado o uso do grafo novamente.

Esta opção permite armazenar um grafo que tenha sido editado por qualquer dos modos de edição apresentados no item 3.2 deste Capítulo.

3.6 - Opções de Algoritmos.

As opções apresentadas a seguir constituem a parte mais importante do sistema SISGRAFO. Estas rotinas permitem aplicar algoritmos específicos da Teoria dos Grafos, tanto para solucionar um dado problema, como obtenção de informações a respeito do grafo com o qual se está trabalhando no momento.

Estas rotinas foram elaboradas de modo inteiramente independente, ou pelo menos procurou-se deixar que isto fosse transparente ao usuário, de tal maneira que, eventualmente, todas elas podem ser solicitadas para um mesmo grafo.

Algumas rotinas, como será apresentado a seguir, devem ser chamadas antes de outras, e se isto não for feito pelo usuário, o sistema se incumbirá de chamá-las. Como de um modo geral isto indica uma conceituação teórica errada, o fato será penalizado com um incremento ao contador de erros.

3.6.1 - Caminho Mínimo (Floyd)

Esta opção foi feita tendo-se como base o algoritmo de FLOYD (3,4) para a determinação dos caminhos mínimos entre todos os pares de vértices de um grafo valorado.

Partindo-se da matriz de valores dos arcos, obtida pela opção do item 3.4.1 deste Capítulo, o algoritmo calcula n sucessivas matrizes de distância, que são modificadas de acordo com a fórmula de recorrência, onde n indica o número de vértices do grafo. O resultado final é dado na última matriz.

O algoritmo obtém também como subproduto a matriz de roteamento do grafo, para as distâncias mínimas, o que é bastante útil para se identificar os vértices pertencentes aos caminhos obtidos.

São admitidos valores negativos para os arcos porém os resultados devem ser analisados com bastante cautela, face a possibilidade da existência de circuitos absorventes.

Esta opção só pode ser chamada após o grafo ter sido valorado.

Sejam:

$V = [v_{ij}]$	matriz de valores
$R = [r_{ij}]$	matriz de roteamento
$D = [d_{ij}]$	matriz de distâncias

Algoritmo:

1. Fazer $D = V$

2. Fazer para $i = 1$ até n e $j = 1$ até n

Se $d_{ij} = \infty$ fazer $r_{ij} = 0$

Caso contrário fazer $r_{ij} = j$

3. Fazer $K = 1$

4. Fazer $i := 1$

5. Fazer para $j = 1$ até $j = n$

$$d_{ij} = \min \left[d_{ij}, d_{ik} + d_{kj} \right]$$

$$r_{ij} = r_{ik} \quad (\text{se houver troca})$$

6. Se $i < n$ fazer $i = i + 1$, ir para o passo 5.

7. Se $k < n$ fazer $k = k + 1$, ir para o passo 4.

Após a execução do algoritmo pode-se pedir então informações sobre a distância entre dois vértices quaisquer, obtendo-se como resposta o valor da distância mínima e também os vértices do caminho que deve ser percorrido para que se obtenha esta distância.

Caso não exista nenhum caminho entre os dois vértices, isto será indicado por uma mensagem.

Os dados de entrada são sempre criticados em função do grafo com o qual se está trabalhando.

3.6.2 - Existência de Circuito

Esta opção utiliza o algoritmo de DEMOUCRON (3) que permite fazer a partição em níveis de um grafo.

O algoritmo determina os vértices pertencentes a cada um dos níveis encontrados. Caso o grafo apresente algum circuito isto será indicado por uma mensagem e, se for possível determinar algum nível antes de se encontrar o primeiro circuito, serão indicados os vértices pertencentes a estes níveis.

Para grafos não orientados o algoritmo indica apenas a existência ou não de ciclos, a partição em níveis não é feita, por não ter sentido neste caso.

Esta opção é bastante útil quando se estiver estudando uma situação cujo grafo associado não deva possuir circuitos, como por exemplo, um problema de caminho máximo (solução de uma rede PERT qualquer).

A rotina parte da determinação sucessiva do semigrau interior de cada vértice, à medida em que este semigrau se torna nulo o vértice é retirado da matriz de adjacência; os vértices retirados em cada iteração constituem cada um dos níveis.

No caso de, em uma das iterações, nenhum semigrau interior ser igual a zero, o grafo possui um circuito.

Para determinação do semigrau interior o algoritmo usa a matriz de adjacência $A = [a_{ij}]$.

Para marcar os v̄rtices que s̄o retirados usa-se um vetor K de n elementos, usa-se ainda duas constantes auxiliares K1 e K2 que assumem os valores 1 ou 0. O vetor S de n elementos cont̄m, a cada iteraç̄o, o semigrau interior dos v̄rtices que n̄o foram retirados anteriormente.

Algoritmo:

1. Inicializar K com zero.

2. Fazer $K1 = 0$

3. Fazer $j = 1$

4. Fazer $S(x_j) = 0$

Se $K_j \neq 0$ ir para o passo 8

5. Fazer $i = 1$

6. Se $K_i \neq 0$ ir para o passo 7

Fazer $S(x_j) = S(x_j) + a_{ij}$

$K1 = 1$

7. Se $i < n$ fazer $i = i + 1$, ir para o passo 6.

8. Se $j < n$ fazer $j = j + 1$, ir para o passo 4.

9. Se $K1 = 0 \Rightarrow$ grafo sem circuito

10. Fazer $K2 = 0$

11. Fazer $j = 1$

12. Se $K_j = 0$ e $S(x_j) = 0$

V̄rtice x_j pertence ao n̄vel.

$K2 = 1$

$K_j = 1$

Caso contr̄rio ir para o passo 13.

13. Se $j < n$ fazer $j = j + 1$, ir para o passo 12.

14. Se $K2 = 0 \Rightarrow$ Grafo com circuito

Caso contrário ir para o passo 2.

3.6.3 - Caminho Máximo

Esta opção foi feita tendo-se como base o algoritmo de FLOYD (3,4) para o cálculo do caminho máximo entre os vértices de um grafo valorado.

Sempre que for chamada esta opção, o algoritmo de Demoucron será usado, para a verificação da existência de circuito, em caso positivo uma mensagem de advertência será dada e o algoritmo de Floyd não será executado.

O algoritmo proposto possui praticamente os mesmos passos do apresentado no item 3.6.1 deste capítulo, para o caminho mínimo; neste caso porém os valores dos arcos inexistentes recebem o valor $-\infty$, e a fórmula de recorrência no passo 5, fica:

$$d_{ij} = \max \left[d_{ij}, d_{ik} + d_{kj} \right]$$

São obtidas as matrizes de distância máxima e de roteamento para obtenção dos vértices pertencentes aos caminhos máximos.

Ao final, podem ser solicitadas informações sobre o caminho máximo entre dois vértices quaisquer do grafo em questão.

3.6.4 - Raio, Diâmetro e Centro

Existem duas opções para o cálculo dos afastamentos, uma para grafos não valorados e outra para os valorados.

No caso dos grafos valorados, além dos valores dos arcos pode-se também fornecer valores ou pesos para os vértices.

São calculados os afastamentos interiores e exteriores, sendo obtidos para os dois casos os seguintes parâmetros: raio, diâmetro, centros e vértices periféricos. São ainda determinados os eventuais centros bidirecionais.

O algoritmo parte do cálculo da matriz de distâncias, $D = [d_{ij}]$

Para efeito de compreensão do algoritmo, de finem-se com índice 0 (zero) as variáveis referentes a separações exteriores e com índice t as referentes a interiores. O duplo índice 0t se refere a separações mistas.

Usa-se ainda um vetor P de n elementos que conterá os pesos fornecidos para os vértices, caso não se deseje fornecer estes valores o vetor P será assumido como tendo o valor unitário.

Algoritmo:

1. Fornecer P.
2. Cálculo de D.
3. Fazer $i = 1$
4. Fazer $j = 1$
5. Fazer: $S_o(x_i) = \max_{x_j \in X} (P(x_j) \cdot d_{ij})$

$$S_t(x_i) = \max_{x_j \in X} (P(x_j) \cdot d_{ji})$$

$$S_{ot}(x_i) = \max_{x_j \in X} [P(x_j)(d_{ij} + d_{ji})]$$

6. Se $j < n$ fazer $j = j + 1$, ir ao passo 5.

7. Se $i < n$ fazer $i = i + 1$, ir ao passo 4.

8. Fazer: $\rho_o(G) = \min_{x_i \in X} [S_o(x_i)]$

$$\rho_t(G) = \min_{x_i \in X} [S_t(x_i)]$$

$$\delta_o(G) = \max_{x_i \in X} [S_o(x_i)]$$

$$\delta_t(G) = \max_{x_i \in X} [S_t(x_i)]$$

$$\rho_{ot}(G) = \min_{x_i \in X} [S_{ot}(x_i)]$$

9. Fazer para $i = 1$ até $i = n$

Se $S_0(x_i) = \rho_0(g) \Rightarrow x_i$ é um centro exterior.

Se $S_0(x_i) = \delta_0(G) \Rightarrow x_i$ é um periférico exterior.

Se $S_t(x_i) = \rho_t(G) \Rightarrow x_i$ é um centro interior.

Se $S_t(x_i) = \delta_t(G) \Rightarrow x_i$ é um periférico interior.

Se $S_{ot}(x_i) = \rho_{ot}(G) \Rightarrow x_i$ é um centro bidirecional.

3.6.5 - Componentes F-Conexas.

Esta opção foi feita usando-se o algoritmo de MALGRANGE (3) para a determinação dos sub-grafos que formam as componentes fortemente conexas de um grafo qualquer.

O algoritmo proposto, por MALGRANGE trabalha partindo de um vértice, que é marcado com (+) e (-); marca-se com (+) todo vértice, não marcado (+); que seja sucessor de um vértice marcado (+), e marca-se com (-) todo vértice, não marcado (-), que seja antecessor de um vértice marcado (-). Quando não se puder marcar mais nenhum vértice, os vértices marcado com (+) e (-) formarão uma componente f-conexa; eliminam-se então estes vértices e recomeça-se o processo com outro vértice.

Para simular a marcação (+) e (-) utilizam-se dois vetores G^+ e G^- de n posições cada um, o vetor G^+ serve de acumulador de quantas vezes cada vértice foi marcado em todo o processo. Existe ainda um vetor MARCA que indica se um determinado vértice está marcado ou não em cada passada.

São usados ainda os conceitos de semigrau exterior: $d^+(x_i)$ e semigrau interior: $d^-(x_i)$.

Algoritmo:

1. Fazer $k = 1$ e $t = 1$
2. Para $i = 1$ até $i = n$
 Calcular $d^+(x_i)$ e $d^-(x_i)$
3. Fazer $G^+(x_k) = 1$ e $AUX = k$
4. Fazer $j = 1$
5. Se $a_{kj} = 0$ ir para o passo 6
 $MARCA(x_j) = x_j$
 $G^+(x_j) = 1$
 $d^+(x_k) = d^+(x_k) - 1$
6. Se $j < n$ fazer $j = j+1$ ir para o passo 5
7. Fazer $j = 1$
8. Se $MARCA(x_j) = 0$ ir para o passo 11
9. Se $d^+(x_j) \leq 0$ fazer $MARCA(x_j) = 0$ ir para o passo 11
10. Fazer $k = j$ e $MARCA(x_j) = 0$ ir para o passo 4
11. Se $j < n$ fazer $j = j+1$ ir para o passo 8
12. Fazer $k = AUX$ e $G^-(x_k) = 1$
13. Fazer $j = 1$
14. Se $a_{jk} = 0$ ir para o passo 15
 $MARCA(x_j) = x_j$
 $G^-(x_j) = 1$
 $d^-(x_k) = d^-(x_k) - 1$

15. Se $j < n$ fazer $j = j + 1$ ir para o passo 14
16. Fazer $j = 1$
17. Se $MARCA(x_j) = 0$ ir para o passo 20
18. Se $d^-(x_j) \leq 0$ fazer $MARCA(x_j) = 0$ ir para o passo 20.
19. Fazer $k = j$ e $MARCA(x_j) \neq 0$ ir para o passo 13.
20. Se $j < n$ fazer $j = j + 1$ ir para o passo 17.
21. Fazer para $i = 1$ até $i = n$

$$G^+(x_i) = G^+(x_i) + G^-(x_i)$$

$$\text{Se } G^+(x_i) = 1 \text{ fazer } G^+(x_i) = 0$$

$$G^-(x_i) = G^+(x_i)$$

22. Fazer $i = 1$
23. Se $G^+(x_i) \neq 0$ ir para o passo 25
24. Fazer $k = i$ e $t = t + 1$
ir para o passo 2
25. Se $i < n$ fazer $i = i + 1$ ir para o passo 23
26. Se $t = 1 \Rightarrow 0$ grafo \bar{e} f-conexo
27. Tem-se então t componentes f-conexas, e cada uma delas será composta pelos vértices x_i que tem o mesmo valor de $G^+(x_i)$.

3.6.6 - Fluxo Máximo

Esta opção foi feita tendo como base o algoritmo de FORD.-FULKERSON (5,12) para obtenção do fluxo máximo em um grafo qualquer.

O algoritmo só é permitido para grafos previamente valorados.

O vértice x_1 será sempre considerado como fonte e o vértice x_n será o sumidouro, assim o fluxo final obtido será dado entre estes dois vértices.

Esta opção utiliza uma proposição feita por Gomory para a realização do algoritmo de Ford-Fulkerson (5).

O algoritmo proposto parte de uma matriz $T(n \times n)$ que recebe inicialmente os dados da matriz de valores $V = [v_{ij}]$. Em T vão sendo calculadas as diferenças entre os valores e os fluxos obtidos, que são colocados em um vetor temporário de n posições, S .

Ao final tem-se em $T = V - T$ a solução ótima, ou seja, os valores dos arcos para que se tenha um fluxo máximo F entre os vértices x_1 e x_n .

Usa-se ainda uma matriz auxiliar para marcação dos vértices, AUX de n por 2.

ALGORÍTMO:

1. Fazer $T = V$
2. Fazer $S = 0$, $AUX = 0$ e $f = 0$
3. Fazer $S_1 = \infty$ e $i = 1$
4. Fazer $j = 2$
5. Se $t_{ij} \leq 0$ e $j = n$ ir para o passo 11
6. Se $S_j \neq 0$ ir para o passo 10
7. Fazer $S_j = \min(S_j, t_{ij})$
 $AUX_{j1} = i$
8. Se $j < i$ fazer $f = 1$

9. Se $j = n$ ir para o passo 15
10. Fazer $j = j + 1$ ir para o passo 5
11. Fazer $AUX_{i2} = 1$
12. Fazer $i = i + 1$
13. Se $i = n$ e $f = 0$ ir para o passo 18
Se $i = n$ e $f \neq 0$ fazer $i = 2$ e $f = 0$
14. Se $S_i > 0$ e $AUX_{i2} = 0$ ir para o passo 4
Caso contrário ir para o passo 12
15. Fazer $t_{ij} = t_{ij} - S_n$
$$t_{ji} = t_{ji} + S_n$$

$$j = i$$
16. Se $j = 1$ ir ao passo 2
17. Fazer $i = AUX_{j1}$ ir para o passo 15
18. O fluxo máximo F será igual a soma de $i = 1$ até n
de t_{i1} . Os valores dos arcos do grafo para se obter
o fluxo máximo estão em $T = V - T$.

Ao findar a execução, esta opção permite que se obtenha, além do valor do fluxo máximo, os valores dos arcos do grafo, não infinitos, existentes em T , indicando ainda quais destes arcos se encontram saturados, ou seja arcos onde o valor (ou capacidade) é igual ao fluxo parcial ($v_{ij} = t_{ij}$).

3.6.7 - Caminho M̄nimo (Dijkstra)

Esta op̄ção foi feita tendo-se como base o algoritmo de DIJKSTRA(3), para o c̄lculo do caminho m̄nimo a partir de um v̄rtice, dito raiz, aos demais v̄rtices de um grafo.

Esta rotina s̄o ̄ v̄lida para grafos previamente valorados.

Inicia-se a matriz de dist̄ncias D com os dados provenientes da valorāo do grafo, que est̄o na matriz V de n por n.

Partindo-se da raiz se obtem os valores (custos) desta at̄ os demais v̄rtices, usando a f̄rmula de recorr̄ncia. Aqueles v̄rtices que possibilitaram o menor custo s̄o marcados e n̄o entram nas pr̄ximas iterāōes.

Usa-se para simular a marcāo, um vetor MARCA de n posīōes.

Os dados entrados como raiz s̄o sempre criticados, quanto a sua coer̄ncia com o grafo que esta sendo usado.

Algoritmo:

1. Fornecer a raiz r
2. Fazer $D = V$ e zerar o vetor MARCA
3. Fazer $i = r$ e $MARCA_r = 1$
4. Fazer $k = 1$
5. Fazer $MIN = \infty$

6. Fazer para $j = 1$ até $j = n$

Se $MARCA_j \neq 1$

$$d_{rj} = \min (d_{rj}, d_{ri} + d_{ij})$$

7. Fazer para $j = 1$ até $j = n$

Se $MARCA_j \neq 1$ ou $MIN > d_{rj}$

$$t = j$$

$$MIN = d_{rj}$$

8. Fazer $MARCA_t = 1$

$$i = t$$

9. Se $k < n - 1$ fazer $k = k + 1$ ir para o passo 5

Ao final da execução do algoritmo tem-se na matriz D as distâncias mínimas da raiz r até todos os vértices do grafo.

A diferença deste algoritmo para o de Floyd apresentado no item 3.6.1 deste capítulo, está no fato de que aquele é executado para todos os vértices e não apenas para a raiz.

A presença deste algoritmo tem a finalidade de permitir uma comparação entre as duas opções para o cálculo do caminho mínimo, e também de atender a problemas que não exigem a determinação de todos os caminhos entre todos os pares de vértices.

3.6.8 - Árvore Parcial Mínima.

Esta opção foi feita tendo-se como base o algoritmo de KRUSKAL (3), para a obtenção de uma árvore geradora de valor mínimo.

Esta rotina só é válida para grafos não orientados e previamente valorados, as rotinas de valoração ou desorientação serão chamadas caso o grafo em uso não possua estas características.

Sendo o grafo $G = (X,U)$ parte-se do grafo $G' = (X,\emptyset)$, onde \emptyset indica a não existência de arcos.

São adicionados a G' os arcos de G a partir do arco de menor valor e seguindo em ordem crescente, desde que a cada novo arco adicionado não seja formado um ciclo com os já existentes. Este processo segue até que se tenha incluído $n - 1$ arcos.

Para verificação da existência de ciclos a rotina proposta utiliza o algoritmo de Demoucron para grafos não orientados, apresentado no item 3.6.2 deste capítulo.

O algoritmo usa três vetores de $\frac{(n-1)n}{2}$ posições para conter os valores dos arcos em ordem crescente:

VMIN: contém os valores

IMIN: contém os valores de i para cada valor de arco.

JMIN: contém os valores de j para cada valor de arco.

Usa-se como partida a matriz de valores,

$$V = [v_{ij}] .$$

A matriz $M = [m_{ij}]$ conterá ao final a árvore parcial mínima na forma booleana.

Algoritmo:

1. Inicializar $M = 0$.
2. Obter $V_{MIN} \Rightarrow v_{ij}$ em ordem crescente
 $IMIN \Rightarrow i$ correspondente a ordem crescente
 $JMIN \Rightarrow j$ correspondente a ordem crescente
3. Fazer $k = 1$
4. Fazer $CONT = 1$
5. Fazer $i = IMIN_k$ e $j = JMIN_k$
6. Fazer $m_{ij} = 1$, $m_{ji} = 1$ e $k = k + 1$
7. Chamar a rotina de Demoucron para verificar-se se com esta inclusão em M , formou algum ciclo.

Se não formou ciclo \Rightarrow ir para o passo 9.

8. Fazer $m_{ij} = 0$, $m_{ji} = 0$ e $CONT = CONT - 1$
9. Se $CONT < n - 1$ fazer $CONT = CONT + 1$ ir para o passo 5.

Uma vez encerrada a execução tem-se na matriz M os arcos da árvore parcial mínima. Pode-se também listar segundo as posições i e j nesta matriz os valores existentes em V_{MIN} que são os valores dos arcos na árvore obtida.

CAPITULO IV - SUGESTÕES DE AMPLIAÇÃO

4.1 - Objetivos

O principal objetivo deste Capítulo é indicar alguns pontos que podem ser alterados, e algumas inclusões de rotinas novas, que possibilitariam um melhor funcionamento do sistema proposto como um todo.

Algumas sugestões são referentes ao próprio programa SISGRAFO; seriam inclusões de rotinas que poderiam entrar na lista de opções, porém outras são orientadas para permitir um melhor controle, tanto administrativo como do próprio arquivo de grafos.

4.2 - Inclusão ou exclusão de arcos e vértices.

Notou-se a grande necessidade de se dispor de meios para alterar a estrutura de um determinado grafo, depois que ele tivesse sido editado. Atualmente, isto só é permitido com uma nova edição de todo o grafo.

Existem quatro objetivos a serem explorados:

(1) Inclusão de arcos.

Este caso é bastante simples, desde que é suficiente colocar-se o valor um na posição desejada da matriz de adjacência.

(2) Inclusão de vértices.

Deve-se fornecer o número de vértices a serem incluídos. Estes vértices serão numerados em sequência à numeração do último vértice. Chamar-se-ia após a rotina para inclusão de arcos e entrar-se-ia com as ligações proporcionadas pelos novos vértices.

(3) Exclusão de arcos

Deve-se, neste caso, colocar o valor zero na posição desejada da matriz de adjacência. Porém é necessário tomar-se cuidado, pois um ou mais vértices podem ficar vazios, ou melhor, com seus semigraus exterior e interior iguais a zero; deste modo estes vértices devem ser eliminados e o grafo deve ser então renumerado, a menos que haja interesse em se trabalhar com o grafo não conexo.

(4) Exclusão de vértices.

Deve-se tornar os semigraus exterior e interior dos vértices desejados iguais a zero. Notar que neste caso torna-se também necessária a renumeração do grafo.

Para todos os casos apresentados acima torna-se necessária a atualização das informações básicas do grafo em questão, tais como: o número de vértices, os semigraus exterior e interior, os sucessores e antecessores de cada vértice e o número de arcos/arestas. O fato de o grafo já ter sido valorado ou não, também deve ser levado em consideração.

4.3 - Sistema de avaliação.

Atualmente o sistema de avaliação está sendo feito pelo incremento do valor unitário ao contador de reentradas. Este sistema poderia ser refeito, para que cada reentrada fosse penalizada de acordo com a gravidade ou desconhecimento teórico atribuído ao erro cometido.

4.4 - Arquivo de informações

Neste arquivo serão armazenadas informações a cada vez que for utilizado o sistema SISGRAFO.

As informações armazenadas, já fazem parte do relatório impresso gerado quando houver solicitação do usuário.

Este arquivo poderá ser útil para uma avaliação do grau de desenvolvimento alcançado pelos usuários do sistema, fazendo-se uso de uma listagem do arquivo em ordem alfabética ou numérica.

As informações armazenadas podem ser:

- (a) nome do usuário,
- (b) número do usuário;
- (c) dia e hora do início da execução,
- (d) dia e hora do fim da execução, e
- (e) número de reentradas efetuadas.

4.5 - Formatação da tela.

Poderia ser feita uma melhor formatação da tela para algumas entradas de dados. Isto visaria possibilitar o fornecimento de vários dados ao mesmo tempo, poupando-se muito tempo na troca de informação.

Evidentemente, isto acarretaria alguns problemas para a crítica dos dados fornecidos e a eventual reentrada de dados incorretos em todo ou em parte.

Um exemplo típico pode ser visto na formatação (figura IV.1), que teria uma tela inicial para a opção um (geração de um grafo aleatoriamente).

Os dados seriam digitados dentro dos colchetes e o cursor (■) saltaria automaticamente para o campo seguinte.

```
***MUDAR DE OPCAO {■ _ _ }  
  
    INFORMACOES SOBRE O GRAFO  
  
***ORIENTACAO { _ _ _ }  
  
***NUMERO DE VERTICES { _ _ }  
  
***NUMERO DE ARCOS/ARESTAS { _ _ _ _ }  
  
***SEMENTE ALEATORIA { _ _ _ _ _ }  
  
***PROBABILIDADE { _ _ _ }  
  
XMT{ _ }
```

Figura IV.1

4.6 - Programas de apoio.

Com algumas pequenas alterações, os programas de apoio poderiam ser adaptados para operação por terminal.

O objetivo seria permitir à pessoa encarregada do gerenciamento do arquivo de grafos uma maior comodidade e segurança para o seu trabalho.

Evidentemente alguns programas, como o LEITOR por exemplo, não devem ser transferidos para o terminal, pois a sua filosofia intrínseca é de permitir a entrada de grafos por cartão.

4.7 - Arquivo de grafos valorados.

Um arquivo de grande interesse seria aquele que permitisse gravar também grafos valorados.

Este arquivo pode conter a mesma estrutura de armazenamento apresentada no Capítulo II, para grafos não valorados, acrescentando-se mais uma informação para os registros de cada vértice, que seria o valor dos arcos partindo dele.

Um programa bastante semelhante ao LEITOR poderia ser usado para permitir o armazenamento destes grafos, que teriam os dados perfurados em cartão.

4.8 - Novos algoritmos.

Existem, na terceira lista de opções, algumas posições livres onde podem ser incluídos novos programas.

Os algoritmos programados devem seguir o mesmo esquema dos já implantados, para que não ocorra incompatibilidade entre as opções existentes.

Em princípio não existe limite para a inclusão de novos algoritmos, podendo-se inclusive abrir novas listas de opções. O limite, para essa expansão diz respeito, apenas, à praticabilidade do uso do sistema, em termos de tempo de resposta, custo etc.

Isto poderia ser evitado pela criação de outras versões do sistema, cada uma delas operando com um número restrito de opções.

APÊNDICE A - GUIA DO USUÁRIO

A.1 - INTRODUÇÃO

Este guia tem a finalidade de permitir ao aluno de um curso sobre Teoria dos Grafos, ou a um usuário qualquer, operar o sistema SISGRAFO proposto com a máxima eficiência.

O sistema foi preparado para operar em terminais de teleprocessamento do computador Burroughs B6700. Para a operação do SISGRAFO é necessário algum conhecimento básico do uso dos terminais (Command And Edit - CANDE), porém procurou-se minimizar a influência deste fato sempre que possível.

Quando o programa entra em módulo de execução, aparecerá na tela o título do sistema e logo abaixo uma mensagem dizendo que a tela está parada para observação. Existe no console uma tecla denominada XMT, que deve ser pressionada neste caso para seguir a execução.

O sistema está montado sob a forma de opções, que devem ser escolhidas pelo usuário. Existem ao todo 31 opções divididas em três listas que são apresentadas na tela. Uma vez escolhida uma delas o sistema executará uma função de controle ou um algoritmo referente a Teoria dos Grafos.

A primeira lista, contendo 11 opções, será sempre apresentada no início, logo após a apresentação do sistema. A princípio pode-se escolher qualquer opção desta lista, porém se nenhum grafo foi ainda editado, esta escolha deve recair sobre uma das opções de edição.

O sistema só permite trabalhar-se com um grafo de cada vez. Caso não se deseje destruir o grafo usado, existe uma opção de armazenamento para colocar o grafo em um arquivo para uso posterior. Esta

gravação é, feita em um arquivo permanente, que pode eventualmente conter outros grafos armazenados em outra oportunidade.

Uma vez que o grafo esteja editado pode-se partir para a execução de qualquer opção, que realize o algoritmo desejado. Usando se o mesmo grafo pode-se chamar quantas opções forem de interesse.

Para maior clareza foi usado neste apêndice um retângulo para representar a tela, as mensagens em seu interior indicam a formatação da tela a cada passo da execução das rotinas.

A.2 - NORMAS GERAIS

A leitura e análise detalhada destas normas permite um melhor manuseio dos recursos existentes no sistema e um maior aproveitamento do tempo de máquina.

1. Para se operar o sistema SISGRAFO é necessário ter-se uma conta autorizada para a utilização do terminal de teleprocessamento do Sistema B6700.
2. Na parte superior direita do console existe uma tecla denominada XMT, que serve de um modo geral para a entrada de qualquer comando via terminal. No módulo de execução do programa SISGRAFO esta tecla deverá ser pressionada: (a) quando esta sigla (XMT) aparecer escrita no video, indicando uma pausa; e (b) sempre que se atender a solicitação de uma entrada de dados.

Esta tecla deve ser pressionada apenas uma vez, o que evita alterações na execução do programa. Em muitas ocasiões, face e sobrecarga na máquina, as respostas se tornam mais lentas; de nada adianta, no entanto, usar novamente o XMT.

3. As paradas da tela são sempre indicadas pela sigla XMT impressa no video. Estas paradas devem ser aproveitadas para a leitura das mensagens ou dados, o mais rápido possível, minimizando assim o tempo de uso da máquina.
4. As mensagens de pedido de dados vem sempre colocadas nas últimas linhas do video, posicionando depois o cursor (■) na primeira linha para a entrada dos dados. Neste apêndice usou-se o sinal () para indicar as posições disponíveis para a entrada dos dados dentro dos colchetes. Por exemplo {■ _ _ _} tem-se quatro posições livres pois a posição do cursor também é utilizável.
5. Os dados só podem ser digitados dentro dos colchetes que aparecem nas primeiras linhas da tela. Caso contrário o programa poderá assumir valores errôneos para os dados fornecidos.

Se algum erro for cometido, antes de apertar a tecla XMT, pode-se posicionar novamente o cursor no início, fazendo-se uso da tecla TAB no console, e entrar com o dado correto.

Quando o campo ou campos, para a entrada de dados estiver totalmente preenchido, ou usando-se a tecla TAB, o cursor se posicionará entre os colchetes próximos à sigla XMT no video, deve-se verificar se o dado ou dados estão corretamente digitados e então pressionar a tecla XMT no console para a entrada de dados.

Por exemplo: {00035} XMT{■}

6. Deve-se procurar observar os limites de tempo de execução e entrada/saída e número de linhas a serem impressas estabelecidos pelo MCP (Master Control Program), para evitar que o sistema operacional forneça uma terminação anormal. 1/
7. Deve-se estar ciente que, sempre que se for editar um novo grafo, ele se sobreporá ao grafo anterior.
8. As respostas às perguntas que pedem a digitação das palavras SIM ou NÃO, podem ser dadas com as próprias palavras ou com os mnemônicos S ou N respectivamente.
9. Usando-se um mesmo grafo pode-se chamar quantas opções se desejar e até mesmo escolher de novo uma opção anteriormente executada.
10. As entradas errôneas são penalizadas, portanto procurar sempre entrar com o dado correto. Por exemplo se o dado é numérico ou literal entrar apenas com números ou letras nas posições e formatos indicados. Maiores esclarecimentos sobre qual o formato dos dados são encontrados nas descrições de cada opção. Existe um contador que é incrementado de um a cada entrada errada, o valor total deste contador, será dado junto com as mensagens de finalização.

1/ No caso do B6700 da UFRJ existem as classes transparentes, e o CANDE forma mais uma classe de processamento com limites estabelecidos.

11. As listagens geradas durante a execução do programa são enviadas para o escaninho indicado na abertura da sessão do terminal; caso não se faça a indicação, as listagens estarão no escaninho que tem o número do terminal que foi usado. 2/
12. Todas as mensagens explicativas ou de solicitação de dados enviadas pelo sistema SISGRAFO são sempre precedidas de três asteriscos (***)).
13. É bastante recomendável que sã se comece a usar o terminal depois de ter um esquema de trabalho bem determinado, evitando-se assim um aumento excessivo do uso da máquina.
14. Ao final, são indicadas no video a data e a hora do início e do fim dos trabalhos, além de uma mensagem de avaliação. Desejando se encerrar a sessão, deve-se entrar com o comando de controle do CANDE, que permite a liberação do terminal (digitar a palavra BYE na primeira linha).

2/ Este é o regime estabelecido para o sistema B6700 instalado no NCE do campus da UFRJ.

A.3 - COMO USAR O PROGRAMA

Uma vez verificado que o terminal esta ligado, entrar com a conta autorizada para a utilização do terminal de teleprocessamento do sistema B6700.

Pode-se executar o programa SISGRAFO de duas maneiras distintas:

1. Colocando-se o programa em uma fila de classes.

Neste caso deve-se elaborar um programa em job symbol usando os comandos de WFL (Work Flow Language) (8) que chamará o SISGRAFO.

Por exemplo 1/ :

```
?BEGIN JOB SISGRAFO;NAME=eee;QUEUE=f;STATION=tt
?MAXPROCTIME=aaa
?MAXIOTIME=bbb
?MAXLINES=cccc
?RUN OBJECT/SISTEMA/SISGRAFO
?FILE FILE6(KIND=DC,MAXRECSIZE=22)
?FILE FILE8(KIND=PRINTER)
?END JOB
```

Onde eee = número inteiro que indica em que escaninho estarão as listagens

f = número inteiro que indica a classe que será colocado o job.

1/ No caso do sistema B6700 da UFRJ, o parâmetro QUEUE não deverá ser indicado, pois o sistema operacional alocará uma classe para o job de acordo com os demais parâmetros.

tt = número inteiro que indica qual terminal está ou
irá operar.

aaa = tempo de processamento (máximo)

bbb = tempo de entrada e saída (máximo)

cccc = número de linhas a serem impressas (máximo)

O usuário deve estar aguardando o início de execução do programa acima, no terminal indicado no parâmetro STATION, pois neste instante o sistema operacional indagará se deseja ou não ir a frente na execução do job.

2. Executar o programa pelo terminal.

Neste caso somente é necessário verificar-se se o programa objeto está disponível para uso na conta indicada.

No caso afirmativo o comando que executará o SISGRAFO tem a seguinte forma:

```
R    OBJECT/SISTEMA/SISGRAFO
```

A.4 - INICIALIZAÇÃO

Quando o programa entrar em módulo de execução, a tela será totalmente limpa e logo após assumirá o seguinte formato:

```
*****
*                                     *
*           SISGRAFO                 *
*                                     *
*           SISTEMA  DIDATICO        *
*                                     *
*   PARA   SOLUCAO DE PROBLEMAS     *
*                                     *
*   ENVOLVENDO   A  TEORIA DOS GRAFOS *
*                                     *
*****

*** A SIGLA XMT INDICA UMA PARADA PARA OBSERVACAO DA TELA
    PARA SEGUIR PRESSIONE A TECLA XMT.
==>XMT
```

Seguindo as instruções o video será limpo e apresentará o seguinte quadro:

```
{ ■ _ _ _ _ _ . . . . . _ _ _ _ _ } { _ _ _ } XMT { _ }
***ENTRE COM O SEU NOME E NUMERO
```

Existem 40 posições para a entrada do nome que poderá conter qualquer caracter alfanumérico ou mesmo especial. As três posições entre os próximos colchetes são para a entrada do número, caso o número tenha menos de três dígitos ele deve ser enquadrado a direita e precedido de brancos ou zeros.

Exemplo:

```
{ JOAO      -      EXERCICIO 1 } {005}      XMT {█}
```

Neste instante deve-se proceder como o indicado na norma 5 apresentada em A.2, uma vez que o cursor se encontre entre os colchetes próximos a XMT, pressiona-se esta tecla no console.

Próximo quadro:

```
{ █ _ _ }                XMT { _ }
```

***DESEJA RELATORIO FINAL IMPRESSO ? SIM OU NAO

SIM - Será impresso um relatório contendo todas as mensagens e informações solicitadas pelo usuário, que foram expostas no video durante a execução do programa. Este relatório estará junto com as mensagens do sistema operacional, no escaninho indicado pelo usuário. Caso o usuário não indique nenhum escaninho as listagens estarão no escaninho que tem o número do terminal em uso. 1/

NAO - Neste caso não será emitido nenhum histórico do que ocorreu no terminal durante a execução do SISGRAFO.

1/ Sistema em vigor no B6700 instalado no campus da UFRJ.

Próximo quadro:

Este quadro apresenta então para escolha, a primeira lista de opções:

{ _ } XMT { _ }

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

Se o grafo ainda não foi editado (não existe), deve-se escolher uma das opções 1,2,3 ou 5, caso contrário uma mensagem de advertência será dada.

Qualquer opção pode ser escolhida, mesmo que não conste da lista apresentada (esta informação vale também para as próximas listas de opções que serão apresentadas mais adiante): Isto evita a obrigação de consulta das demais listas, sempre que já se saiba previamente o número da opção desejada.

Sempre que for solicitada a escolha de uma nova opção, o programa apresentará esta primeira lista.

De acordo com a opção escolhida serão impressas então algumas informações sobre ela, como: algoritmo utilizado, restrições, etc. Maiores detalhes serão dados na apresentação de cada uma das opções.

Existe ainda neste instante, uma oportunidade de se mudar de opção (exceção feita a opção 0), pela apresentação do seguinte quadro:

<input type="checkbox"/> { } XMT { }
***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha

NAO - Continuará a execução da opção escolhida.

Esta pergunta será sempre formulada quando se escolhe uma nova opção.

Após se ter começado a executar uma opção, não se pode mais interromper o seu processamento (a menos que se usem comandos do CANDE, como: ?DS ou ?BRK) (7).

A melhor maneira é não interromper o processamento, o que acarretaria uma terminação anormal, e deixar que a opção seja executada, procurando responder as perguntas visando a sua finalização o mais rápido possível, até que se chegue à pergunta relativa à execução, ou não de uma nova opção.

A.5 - OPÇÕES

Neste ítem serão analisadas todas as opções existentes no sistema SISGRAFO. Estas opções são escolhidas pelo seu respectivo número; para facilitar a sua localização, elas foram aqui colocadas nesta ordem numérica.

As paradas da tela para observação são indicadas pela sigla XMT, que aparecerá ao final das mensagens e do lado esquerdo do video. Como indica a norma 2 em A.2, para prosseguir deve-se pressionar a tecla XMT no console apenas uma vez.

OPCAO LIDA = 0

Esta opção permite encerrar a execução do programa a qualquer tempo.

Uma vez escolhida, esta opção dará a seguinte mensagem.

```
*** DEVIDO A OPCA0 = 0 OS TRABALHOS SERAO INTERROMPIDOS  
  
==>XMT
```

Em seguida, a tela poderá apresentar uma das duas disposições abaixo, contendo as mensagens de finalização.

(1) Se não houver nenhuma reentrada.

*** PARABENS, VOCE NAO NECESSITOU NENHUMA REENTRADA

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA

APOS ENTRE BYE

*** INICIO DIA 25/10/79 HORA 20:12

*** TERMINO DIA 25/10/79 HORA 21:15

(2) Se houve alguma reentrada (por exemplo 5).

*** VOCE EXECUTOU 5 REENTRADAS

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA

APOS ENTRE BYE

*** INICIO DIA 10/11/79 HORA 22:42

*** TERMINO DIA 10/11/79 HORA 23:52

*** OPCAO LIDA = 1

GERA UM GRAFO ALEATORIAMENTE

PODE GERAR GRAFOS ORDENADOS OU NAO

O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)

A SEMENTE PARA GERACAO PODE SER QUALQUER NUMERO

A PROBABILIDADE DEVE SER > ZERO E <= CEM

PODE-SE LIMITAR O NUMERO DE ARCOS DESEJADOS

ESTA E UMA ROTINA DE EDICAO DE UM GRAFO

PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR

APOS A GERACAO JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE

OS SUCESSORES DE CADA VERTICE

==>XMT

Informações solicitadas:

a . Escolher outra opção

{ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 1.

b. Orientação do grafo.

{ <input type="checkbox"/> _ _ }	XMT { _ }
***O GRAFO E ORIENTADO? SIM OU NAO	

SIM - Se desejar gerar um grafo orientado

NAO - Caso contrário

c. Número de vértices do grafo.

{ <input type="checkbox"/> _ }	XMT { _ }
*** ENTRE COM O NUMERO DE VERTICES DO GRAFO, MAXIMO 40	

Seja n o número de vértices do grafo.

São serão aceitos valores tais que:

$$0 < n \leq 40$$

Enquadrar o número à direita, precedido de branco ou zero, se necessário:

Entradas válidas:

{ 5 } XMT { }

{ 09 } XMT { }

{ 32 } XMT { }

d. Número de arcos/arestas do grafo.

{ _ _ _ }

XMT{ }

***ENTRE COM O NUMERO DE ARCOS/ARESTAS DO GRAFO
CASO NAO DESEJE LIMITAR ENTRE COM ZERO

Seja n o número de vértices e m o número de arcos/arestas do grafo.

São serão aceitos valores tais que:

(a) grafo orientado

$$0 \leq m \leq n(n-1)$$

(b) grafo não orientado

$$0 \leq m \leq \frac{n(n-1)}{2}$$

Enquadrar o número à direita, precedido de brancos ou zeros, se necessário.

Caso se forneça o valor zero, será gerado um grafo com o número de arcos/arestas de acordo com a probabilidade fornecida adiante.

Para valores maiores que zero, o grafo gerado poderá ter aquele número de arcos/arestas ou menos.

Entradas válidas:

Se $n = 10$ e grafo orientado

{0090}	XMT{ <input type="checkbox"/> }
{ 45}	XMT{ <input type="checkbox"/> }
{ 0}	XMT{ <input type="checkbox"/> }
{ }	XMT{ } assume o valor zero

e. Semente aleatória

{ ■ _ _ _ _ }	XMT { _ }
***ENTRE COM A SEMENTE ALEATÓRIA	

Serão aceitos quaisquer números; o número será lido com duas casas decimais, a menos que se especifique a posição do ponto decimal.

Entradas válidas:

- | | | |
|----------|---------|------------------------|
| {569.56} | XMT {■} | |
| {003.42} | XMT {■} | |
| {47346.} | XMT {■} | |
| { 40832} | XMT {■} | valor interno = 408.32 |

f. Probabilidade desejada

{ ■ _ _ }	XMT { _ }
***ENTRE COM A PROBABILIDADE DA EXISTENCIA DE ARCOS/ARESTAS	

A probabilidade \bar{e} entrada em porcentagem seja p a probabilidade:

São serão aceitos valores tais que:

$$0 < p \leq 100$$

Se o número de arcos/arestas escolhidos no item d for maior que zero, não será pedida esta probabilidade e seu valor será dado por:

(a) grafos orientados

$$p = \frac{m}{n (n - 1)}$$

(b) grafos não orientados

$$p = \frac{2 m}{n (n - 1)}$$

Enquadrar o número à direita precedido de brancos ou zeros.

Entradas válidas:

{050} XMT {■}

{100} XMT {■}

{ 35} XMT {■}

g. Após a geração

Tendo gerado o grafo são dadas informações sobre os arcos/arestas, vértices e orientação, como no exemplo abaixo.

*** O GRAFO GERADO TEM:

14 ARCOS/ARESTAS

6 VERTICES

E ORIENTADO

==> XMT.

h. Nova geração

<input type="checkbox"/> _ _ }	XMT { _ }
*** PREFERE GERAR UM GRAFO DIFERENTE? SIM OU NAO	

SIM - Reiniciará a rotina de geração, no item b.

NAO - Segue a execução

i. Matriz de adjacência

<input type="checkbox"/> _ _ }	XMT { _ }
*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA? SIM OU NAO	
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES	

SIM - Mostrará no video a matriz de adjacência na forma booleana do grafo gerado. Se o grafo tiver mais de 15 vértices, a matriz não será mostrada e será dada uma mensagem de advertência. Neste caso pode-se solicitar a impressão da matriz de adjacência, escolhendo depois a opção número 8.

NAO - Segue a execução

j. Número de sucessores.

```
{ ■ _ _ }           XMT { _ }
```

***DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES? SIM OU NAO

SIM - Apresenta para todos os vértices o semigrau exterior de cada um.

NAO - Segue a execução

k. Os sucessores

```
{ ■ _ _ }           XMT { _ }
```

*** DESEJA SABER OS SUCESSORES DOS VERTICES? SIM OU NAO

SIM - Mostra na tela para cada vértice, além de seu semigrau exterior, os seus sucessores diretos. Se algum vértice não tem sucessores isto será indicado. São apresentados na tela apenas três vértices de cada vez, para observar os próximos três, pressionar apenas uma vez a tecla XMT no console, e assim sucessivamente até o final.

NAO - Segue a execução

1. Nova Opção.

{█ _ _}	XMT { _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçãõ.

*** OPCAO LIDA = 2

CHAMA UM GRAFO ARMAZENADO ANTERIORMENTE
E NECESSARIO SABER O NUMERO DO GRAFO A SER USADO
TODO GRAFO ARMAZENADO E ORIENTADO
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR
APOS A LEITURA JA ESTAO A DISPOSICAO:
A MATRIZ DE ADJACENCIA
O SEMIGRAU EXTERIOR DE CADA VERTICE
OS SUCESSORES DE CADA VERTICE

==> XMT

Informações solicitadas:

a. Escolher outra opção.

{ __ }

XMT{ _ }

*** PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 2.

b. Número do grafo escolhido.

```
{█_ _}          XMT { _ }

*** ENTRE COM O NUMERO DO GRAFO PADRONIZADO
```

Deve-se indicar o número do grafo escolhido existente no arquivo de grafos.

Caso o número do grafo solicitado não seja encontrado no arquivo, será dada uma mensagem de advertência e poderá ser escolhido outro grafo; são admitidas três tentativas de entrada de um número válido de grafo, isto é, de um grafo existente no arquivo.

Enquadrar o número à direita, precedido de brancos ou zeros.

Entradas válidas:

```
{005}          XMT {█ }
{ 3}          XMT {█ }
```

c. Após a leitura.

Uma vez lido o grafo, serão dadas informações sobre os arcos/arestas, vértices e orientação, como no exemplo abaixo.

```
*** O GRAFO CHAMADO TEM:
    19  ARCOS/ARESTAS
    13  VERTICES
        E ORIENTADO

==>XMT
```

Em princípio, todos os grafos armazenados são orientados. Pode-se chamar uma opção própria para desorientação, se este for o caso.

d. Novo grafo armazenado

{█ _ _}	XMT { }
*** PREFERE ESCOLHER OUTRO GRAFO PADRONIZADO? SIM OU NAO	

SIM - Reiniciará a rotina para chamar um outro grafo, no item b.

NAO - Segue a execução

e. Matriz de adjacência

{█ _ _}	XMT { }
***DESEJA OBSERVAR A MATRIZ DE ADJACENCIA? SIM OU NAO	
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES	

SIM - Mostrará no vídeo a matriz de adjacência na forma booleana do grafo chamado. Se o grafo tiver mais de 15 vértices, a matriz não será mostrada e será dada uma mensagem de advertência. Neste caso pode-se solicitar a impressão da matriz de adjacência, escolhendo depois a opção número 8.

NAO - Segue a execução

f. Número de sucessores

```
{█ _ _} XMT {}  
***DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES? SIM OU NAO
```

SIM - Apresenta para todos os v̄rtices o semigrau exterior de cada um
NAO - Segue execu~ao

g. Os sucessores

```
{█ _ _} XMT{ _ }  
*** DESEJA SABER OS SUCESSORES DOS VERTICES? SIM OU NAO
```

SIM - Mostra na tela, para cada v̄rtice, al̄m de seu semigrau exterior os seus sucessores diretos. Se algum v̄rtice n̄o tem sucessores, isto ser̄a indicado. S̄ao apresentados na tela apenas tr̄es v̄rtices de cada vez; para observar os pr̄oximos tr̄es, pressionar ape- nas uma vez a tecla XMT no console, e assim sucessivamente at̄e o final.

NAO - segue a execu~ao

h. Nova op~ao

```
{█ _ _} XMT { _ }  
*** DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO
```

SIM - Mostrar̄a a primeira lista de op~oes para escolha
NAO - Dar̄a as mensagens de finaliza~ao

*** OPCAO LIDA = 3

LE UM GRAFO FORNECIDO PELO TERMINAL

O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)

NAO E ACONSELHAVEL O USO PARA GRAFOS GRANDES (> 15 VERTICES)

PARA CADA VERTICE FORNECA TODOS OS SUCESSORES

SE O VERTICE NAO TEM SUCESSORES ENTRE ZERO

PODE-SE LER GRAFO ORIENTADO OU NAO

PARA OS NAO ORIENTADOS PODE-SE ENTRAR SO COM OS SUCESSORES:

DO TRIANGULO INFERIOR DA MATRIZ DE ADJACENCIA

A MATRIZ E SIMETRIZADA EM RELACAO AO TRIANGULO INFERIOR

ESTA E UMA ROTINA DE EDICAO DE UM GRAFO

PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR

APOS A LEITURA JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE

OS SUCESSORES DE CADA VERTICE

==> XMT

Informações solicitadas.

a. Escolher outra opção.

{ ■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 3.

b. Orientação do grafo

{█ _ _}	XMT { }
*** O GRAFO E ORIENTADO? SIM OU NAO	

SIM - Se o grafo a ser lido é orientado

NAO - Caso contrário

c. Número de vértices do grafo.

{█ _ _}	XMT { }
*** ENTRE COM O NUMERO DE VERTICES DO GRAFO, MAXIMO 40	

Seja n o número de vértices do grafo.

São serão aceitos valores tais que:

$$0 < n \leq 40$$

Enquadrar o número à direita precedido de branco ou zero.

Entradas válidas:

{12} XMT {█ }

{ 3} XMT {█ }

{06} XMT {█ }

d. Semigrau exterior e sucessores.

Seguindo a ordem numérica ascendente dos vértices, a partir de 1, serão solicitados o semigrau exterior e quais são os sucessores de cada um.

{█_} XMT {_}

*** ENTRE COM O SEMIGRAU EXTERIOR DO VERTICE i

Onde i indica o número do vértice para o qual a informação é solicitada.

Seja n o número de vértices do grafo e m_i o número de sucessores do vértice i.

São serão aceitos valores tais que:

$$0 \leq m_i \leq n$$

Caso o vértice não tenha sucessores entrar com o valor zero.

Enquadrar o número à direita, procedido de branco ou zero, caso necessário.

Entradas válidas:

{12} XMT{█}

{03} XMT{█}

{ 9} XMT{█}

{ } XMT{█} assume o valor zero

A seguir vem a solicitação dos sucessores.

Dependendo de quantos sucessores tem o vértice em questão, serão apresentadas tantas lacunas entre colchetes quantas

necessárias para a entrada dos dados.

Serão apresentadas dez posições em cada linha para a entrada de valores, assim se um vértice tiver mais de dez sucessores, serão formatadas duas ou mais linhas.

```
{█ } { } { } { } { } . . . . . { }  
XMT{ }  
*** ENTRE COM OS  $m_i$  SUCESSORES DO VERTICE  $i$ 
```

A entrada dos sucessores poderá ser em qualquer ordem.
Enquadrar os números à direita, precedidos de branco ou zero.

Entradas válidas:

```
{03} { 5} { 2} {10} {06} {08}
```

```
XMT {█}
```

No caso de um vértice com mais de dez sucessores:

```
{02} {03} {05} {09} {11} {12} {16} {13} {14} {08}
```

```
{15} {17} {19}
```

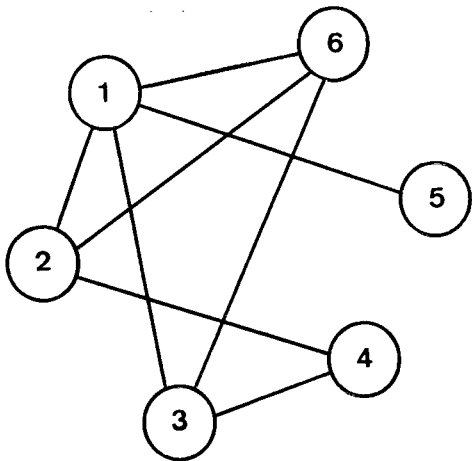
```
XMT{█}
```

Observação: Simplificação para grafos não orientados.

Neste caso pode-se entrar apenas com os sucessores do triângulo inferior da matriz de adjacência; o semigrau exterior deverá estar de acordo.

A matriz de adjacência do grafo será gerada internamente simetrizando-se seu triângulo inferior.

EXEMPLO:



MATRIZ DE ADJACÊNCIA

	1	2	3	4	5	6
1	0	1	1	0	1	1
2	1	0	0	1	0	1
3	1	0	0	1	0	1
4	0	1	1	0	0	0
5	1	0	0	0	0	0
6	1	1	1	0	0	0

Entrada reduzida:

Vértice	Semigrau	Sucessores
1	0	-
2	1	1
3	1	1
4	2	2 e 3
5	1	1
6	3	1,2 e 3

e. Reentrada do vértice

Após a entrada de cada vértice, será apresentada na tela a informação lida do semigrau e dos sucessores. Caso não se esteja de acordo poder-se-á entrar novamente.

{■ _ _ }

XMT { _ }

*** DESEJA ENTRAR COM O VERTICE NOVAMENTE? SIM OU NAO

SIM - Retorna ao item d, para nova entrada dos dados referentes ao v̄ertice i

NAO - Retorna ao item d, para a entrada dos dados referentes ao pr̄oximo v̄ertice (i + 1), ou continuaç̄ão da execuç̄ão, se j̄a entraram todos os v̄ertices.

f. Ap̄os a leitura

Uma vez lidos todos os dados sobre o grafo s̄ao dadas informaç̄ões sobre os arcos/arestas, v̄ertices e orientaç̄ão, como no exemplo abaixo.

```
*** O GRAFO LIDO TEM:  
    15 ARCOS/ARESTAS  
    10 VERTICES  
    NAO E ORIENTADO  
  
==>XMT
```

g. Ler outro grafo

```
{█_ _}          XMT {_}  
  
*** PREFERE LER OUTRO GRAFO? SIM OU NAO
```

SIM - Reiniciar̄a a rotina para leitura de outro grafo, no item b.

NAO - Segue a execuç̄ão

h. Matriz de adjacência

<input type="checkbox"/> _ _ }	XMT{ _ }
*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA? SIM OU NAO	
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES	

SIM - Mostrarã no video a matriz de adjacência na forma booleana do grafo lido. Se o grafo tiver mais de 15 vértices, a matriz não será mostrada e será dada uma mensagem de advertencia. Neste caso pode-se solicitar e impressão da matriz de adjacência, escolhendo depois a opção número 8.

NAO - Segue a execução

i. Número de sucessores

<input type="checkbox"/> _ _ }	XMT { _ }
*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES? SIM OU NAO	

SIM - Apresenta, para todos os vértices, o semigrau exterior de cada um.

NAO - Segue a execução

j. Os sucessores

```
{█ _ _}                                XMT { _ }
```

***DESEJA SABER OS SUCESSORES DOS VERTICES? SIM OU NAO

SIM - Mostra na tela para cada vértice, além de seu semigrau exterior, os seus sucessores diretos. se algum vértice não tem sucessores, isto será indicado. São apresentados na tela apenas três vértices de cada vez; para observar os próximos três, pressionar apenas uma vez a tecla XMT no console, seguindo assim até o final.

NAO - segue a execução

k. Nova opção

```
{█ _ _}                                XMT { _ }
```

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha
NAO - Dará as mensagens de finalização.

***OPCAO LIDA = 4

ARMAZENA UM GRAFO PARA USO POSTERIOR

PODE ARMAZENAR GRAFO LIDO, GERADO OU PARTE LIDO PARTE GERADO

E NECESSARIO CONHECER A SENHA AUTORIZADA

A CADA GRAFO ARMAZENADO E DADO UM NUMERO SEQUENCIAL

EXCEDENDO A CAPACIDADE DO ARQUIVO O GRAFO NAO E ARMAZENADO

AO FINAL E INDICADO O NUMERO QUE RECEBEU O SEU GRAFO

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{█ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 4, somente se o grafo
já foi editado.

b. Senha autorizada

█

***PARA GRAVAR ENTRE COM A SENHA AUTORIZADA

A senha deve ser entrada na primeira linha da tela.

Existem três chances para entrar com a senha correta.

Esta senha será fornecida pela pessoa responsável pelo sistema SISGRAFO.

Caso não se consiga entrar com a senha exata em três chances, será dada uma mensagem de advertência e entrará no vídeo a primeira lista de opções para uma nova escolha.

Será dada uma mensagem dizendo que número recebeu o grafo que acabou de ser armazenado e também uma indicação de qual foi a última posição gravada no arquivo; esta informação deverá ser passada para a pessoa responsável pelo sistema SISGRAFO.

Pode ocorrer que não haja espaço suficiente no arquivo; neste caso, uma mensagem indicativa será dada, além de informações sobre os apontadores para manutenção do arquivo.

c. Nova opção

<input type="checkbox"/> _ _ }	XMT { _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização

***OPCAO LIDA = 5

LE UM GRAFO PELO TERMINAL DEPOIS GERA MAIS ARCOS
O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)
NAO E ACONSELHABEL PARA GRAFOS COM MUITOS ARCOS
INDIQUE PARA CADA VERTICE SE ELE TEM OU NAO SUCESSORES
PARA OS NAO ORIENTADOS PODE-SE ENTRAR SO COM OS SUCESSORES:
DO TRIANGULO INFERIOR DA MATRIZ DE ADJACENCIA
A MATRIZ E SIMETRIZADA EM RELACAO AO TRIANGULO INFERIOR
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR
APOS A GERACAO JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE (TOTAL)

OS SUCESSORES DE CADA VERTICE (TOTAL)

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{█ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execuãõ da opção 5.

b. Orientação do grafo

{█ _ _ }	XMT{ _ }
***O GRAFO E ORIENTADO? SIM OU NAO	

SIM - Se o grafo a ser montado é orientado

NAO - Caso contrário

c. Número de vértices do grafo

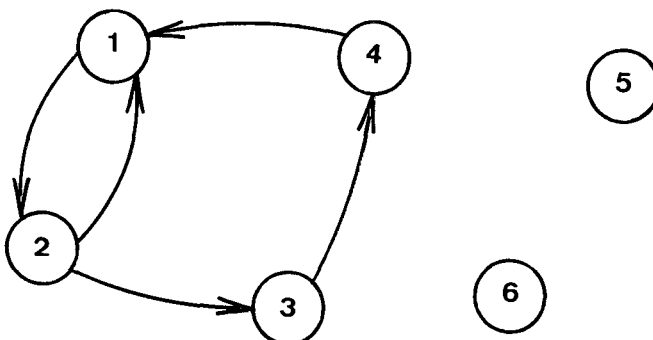
{█ _ }	XMT{ _ }
***ENTRE COM O NUMERO DE VERTICES DO GRAFO, MAXIMO 40	

Seja n o número de vértices do grafo.

São serão aceitos valores tais que:

$$0 < n \leq 40$$

Fornecer neste instante o número total de vértices do grafo. Mesmo que algum vértice não tenha arestas chegando ou partindo dele, deve ser incluído como no exemplo.



Neste caso, $n=6$ e não $n=4$.

Enquadrar o número à direita, precedido de branco ou zero.

Entradas válidas:

{31} XMT{■}

{ 5} XMT{■}

{04} XMT{■}

d. Semigrau exterior e sucessores.

Seguindo a ordem numérica ascendente dos vértices, a partir de 1, serão solicitados o semigrau exterior e quais são os sucessores de cada um.

{■}	XMT{ }
***ENTRE COM O SEMIGRAU EXTERIOR DO VERTICE i	

Onde i indica o número do vértice para o qual a informação é solicitada.

Seja n o número de vértices do grafo e m_i o número de sucessores do vértice i .

São serão aceitos valores tais que:

$$0 \leq m_i \leq n$$

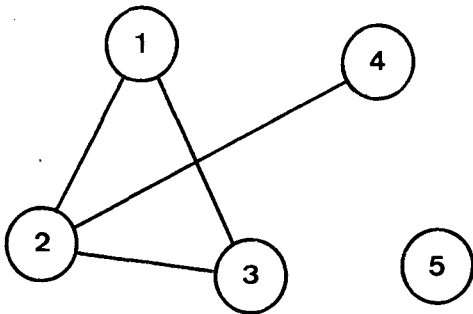
Caso o vértice não tenha sucessores entrar com o valor zero.

Observação: Simplificação para grafos não orientados.

Neste caso, pode-se entrar apenas com os sucessores do triângulo inferior da matriz de adjacência; o semigrau exterior deverá estar em acordo.

A matriz de adjacência do grafo será gerada internamente, simetrizando-se seu triângulo inferior.

Exemplo:



Matriz de Adjacência

	1	2	3	4	5
1	0	1	1	0	0
2	1	0	1	1	0
3	1	1	0	0	0
4	0	1	0	0	0
5	0	0	0	0	0

Entrada Reduzida:

Vértice	Semigrau	Sucessores
1	0	-
2	1	1
3	2	1 e 2
4	1	2
5	0	-

e. Reentrada do vértice

Após a entrada de cada vértice, será apresentada na tela a informação lida do semigrau e dos sucessores. Caso não se esteja de acordo poder-se-á entrar novamente.

```
{█ _ _} XMT{ _ }
```

***DESEJA ENTRAR COM O VERTICE NOVAMENTE? SIM OU NAO

SIM - Retorna ao item d, para a entrada dos dados novamente, referentes ao vértices i.

NAO - Retorna ao item d, para a entrada dos dados referentes ao próximo vértice (i+1), ou continuação da execução, se já entraram todos os vértices.

f. Após a leitura.

Uma vez fornecidos os dados referentes à parte do grafo lida, são dadas informações sobre os arcos/arestas, vértices e orientação, como no exemplo abaixo.

```
***O GRAFO LIDO TEM:  
4  ARCOS/ARESTAS  
5  VERTICES  
   NAO E ORIENTADO  
  
==> XMT
```

g. Ler outro grafo.

{█_ _}	XMT{ _ }
***PREFERE LER OUTRO GRAFO? SIM OU NAO	

SIM - Reiniciará a rotina para a leitura de outro grafo, no item b.

NAO - Segue a execução.

h. Matriz de adjacência

{█_ _}	XMT{ _ }
***DESEJA OBSERVAR A MATRIZ DE ADJACENCIA? SIM OU NAO	
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES	

SIM - Mostrará no video a matriz de adjacência na forma booleana, do grafo lido. Se o grafo tiver mais de 15 v̄ertices, a matriz n̄o ser̄a mostrada e ser̄a dada uma mensagem de advertência. Neste caso pode-se solicitar a impress̄o da matriz de adjacência, escolhendo depois a opç̄o n̄mero 8.

NAO - Segue a execução.

i. Número de sucessores

```
{█ _ _} XMT{ _ }  
  
***DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES? SIM OU NAO
```

SIM - Apresenta para todos os vértices o semigrau exterior de cada um.

NAO - Segue a execução.

j. Os sucessores

```
{█ _ _} XMT{ _ }  
  
***DESEJA SABER OS SUCESSORES DOS VERTICES? SIM OU NAO
```

SIM - Mostra na tela, para cada vértice, além de seu semigrau exterior, os seus sucessores diretos. Se algum vértice não tem sucessores, isto será indicado. São apresentados na tela apenas três vértices de cada vez; para observar os próximos três, pressionar apenas uma vez a tecla XMT no console, seguindo assim até o final.

NAO - Segue a execução.

Enquadrar o número à direita, precedido de branco ou zero.

Entradas válidas:

- {05} XMT{█}
- {11} XMT{█}
- { 0} XMT{█}
- { } XMT{█} assume o valor zero.

A seguir vem a solicitação dos sucessores.

Dependendo de quantos sucessores tem o vértice em questão, serão apresentadas tantas lacunas entre colchetes quantas necessárias para a entrada dos dados.

Serão apresentadas dez posições em cada linha para entrada dos valores; assim se um vértice tiver mais de 10 sucessores, serão formatadas duas ou mais linhas.

```
{█} { } { } { } { } .....{ }  
XMT{ }  
***ENTRE COM OS mi SUCESSORES DO VERTICE i
```

A entrada dos sucessores poderá ser em qualquer ordem.

Enquadrar o número à direita, precedido de branco ou zero.

Entradas válidas:

{02} {04} { 5} {12} {11}

XMT {█}

No caso de um vértice com mais de dez sucessores:

{01} {03} {05} {08} {10} {15} {18} {19} {17} {20}

{21} {22} {09} {23}

XMT { }

K. Semente aleatória.

{█ _ _ _ _ }	XMT{ _ }
***ENTRE COM A SEMENTE ALEATORIA	

São aceitos quaisquer números, o número será lido com duas casas decimais, a menos que se especifique a posição do ponto.

Entradas válidas:

{361.15}	XMT{█ }	
{ 33.52}	XMT{█ }	
{1.1 }	XMT{█ }	
{004597}	XMT{█ }	valor interno = 45.97

1. Probabilidade desejada

{█ _ _ }	XMT{ _ }
***ENTRE COM A PROBABILIDADE DA EXISTENCIA DE ARCOS/ARESTAS	

A probabilidade é entrada em percentagem.

Seja p a probabilidade.

São serão aceitos valores tais que:

$$0 < p \leq 100$$

Enquadrar o número à direita, precedido de brancos ou zeros. Os números devem ser inteiros, ou seja não podem conter o ponto decimal.

Entradas válidas:

{035}	XMT{█}
{ 5}	XMT{█}
{ 82}	XMT{█}

m. Após a geração.

Tendo gerado os arcos/arestas adicionais ao grafo lido, são dadas as informações referentes ao grafo final obtido, como no exemplo abaixo.

```
***O GRAFO FINAL TEM:  
    7  ARCOS/ARESTAS  
    5  VERTICES  
      NAO E ORIENTADO  
==>XMT
```

n. Construir outro grafo

```
{█ _ } XMT { _ }  
  
***PREFERE MONTAR OUTRO GRAFO? SIM OU NAO
```

SIM - Reiniciará a rotina a partir do item b, para a montagem de um novo grafo.

NAO - Segue a execução

o. Informações sobre o grafo.

Volta a executar os itens h, i e j, porém agora com o grafo completo.

Estes itens se referem a matriz de adjacência, semigrau exterior e sucessores.

p. Nova opção

<input type="checkbox"/> _ _ }	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

*** OPCAO LIDA = 6

MOSTRA NA TELA OS SUCESSORES DE CADA VERTICE

SAO LISTADOS NA TELA TODOS OS VERTICES COM:

O SEMIGRAU EXTERIOR

QUAIS SAO OS SUCESSORES DE CADA UM

==>XMT

Informações solicitadas:

a. Escolher outra opção.

█ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 6, somente se o grafo já foi editado.

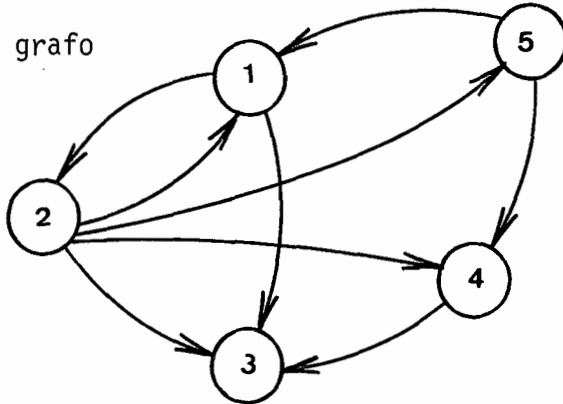
b. Disposição da tela.

Apresenta, para três vértices de cada vez, as informações sobre o semigrau exterior e quais são os sucessores.

Para observar os demais vértices deve-se pressionar a tecla XMT, apenas uma vez, após a apresentação de cada série de três vértices.

Exemplo:

Seja o grafo



Primeira tela:

```
***SUCESSORES DOS VERTICES
***VERTICE 1 SEMIGRAU EXTERIOR 2
SUCESSORES 2 3
***VERTICE 2 SEMIGRAU EXTERIOR 4
SUCESSORES 1 3 4 5
***VERTICE 3 NAO TEM SUCESSORES

== > XMT
```

Segunda tela:

```
***SUCESSORES DOS VERTICES
***VERTICE 4 SEMIGRAU EXTERIOR 1
SUCESSORES 3
***VERTICE 5 SEMIGRAU EXTERIOR 2
SUCESSORES 1 4

== > XMT
```

c. Nova opção

{■ _ _ }

XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

***OPCAO LIDA = 7

MOSTRA O SEMIGRAU EXTERIOR DE CADA VERTICE

SAO LISTADOS NA TELA TODOS OS VERTICES COM:

O SEMIGRAU EXTERIOR DE CADA UM.

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{■ _ _}

XMT { _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

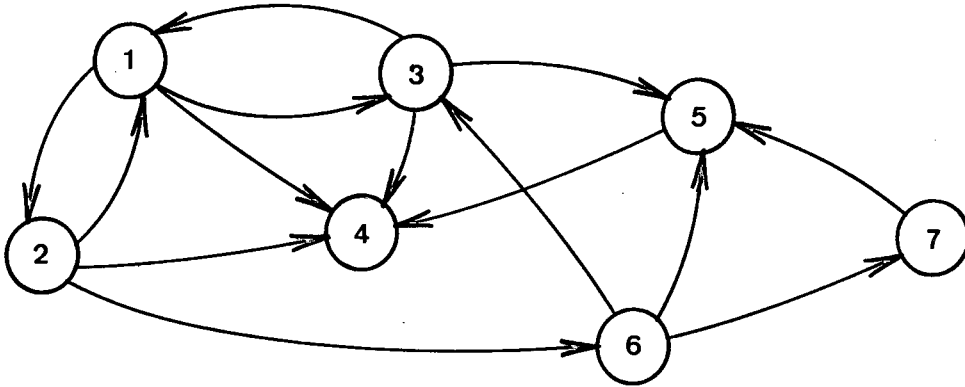
NAO - Continuarã a execução da opção 7, somente se o grafo já foi editado.

b. Disposição da tela.

Apresenta para cada vértice somente o seu semigrau exterior.

Exemplo:

Seja o grafo:



***TOTAL DE SUCESSORES DOS VERTICES

VERT	SUC	VERT	SUC	VERT	SUC	VERT	SUC
1	3	3	3	5	1	7	1
2	3	4	0	6	3		

==> XMT.

Para os grafos não orientados, tudo se passa como se tivessem orientação nos dois sentidos.

c. Nova opção.

{ ___ }

XMT{ ___ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

***OPCAO LIDA = 8

SOLTA UMA LISTAGEM DA MATRIZ DE ADJACENCIA

IMPRIME A MATRIZ DE ADJACENCIA PARA QUALQUER GRAFO

USADA QUANDO NAO SE PODE OBSERVAR A MATRIZ NA TELA

PRINCIPALMENTE GRAFOS COM MAIS DE 15 VERTICES

O ESCANINHO DA LISTAGEM TEM O NUMERO DO TERMINAL USADO

==> XMT .

Informações solicitadas.

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha

NAO - Continuarã a execução da opção 8, somente se o grafo já foi editado.

b. Disposição da tela.

Esta opção é usada quando se deseja observar a matriz de adjacência, porém o grafo tem mais de 15 vértices, e isto não pode ser feito no video.

Serã impressa a matriz em sua representação booleana.

Após a impressão será dada a mensagem na tela:

```
***FORAM IMPRESSOS OS RESULTADOS PEDIDOS  
  
O ESCANINHO TEM O NUMERO DO TERMINAL EM USO  
  
==>XMT
```

Caso na abertura da sessão do terminal se especifique o escaninho desejado, as listagens serão dirigidas para ele e não para o de número do terminal em uso.

c. Nova opção.

```
{█ _ _ } XMT{ _ }  
  
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO
```

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

***OPCAO LIDA = 9

MOSTRA NA TELA A MATRIZ DE ADJACENCIA
SO PODE SER USADA PARA GRAFOS COM ATE 15 VERTICES
PARA GRAFOS MAIORES USAR A OPCA0 DE IMPRIMIR A MATRIZ
==> XMT

Informações solicitadas:

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execuçaõ da opção 9, somente se o grafo já foi editado.

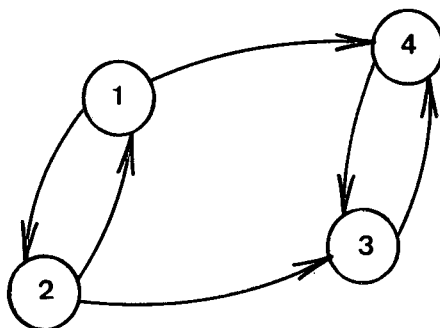
b. Disposiçaõ da tela.

Se eventualmente for solicitado para observar a matriz de um grafo com mais de 15 vértices, será dada uma mensagem de advertência. Neste caso pode-se executar a opção anterior (número 8) para a impressão da matriz de adjacência.

Apresenta a matriz em sua representaçaõ booleana.

Exemplo:

Seja o grafo



***MATRIZ DE ADJACENCIA DO GRAFO 4x4

	1	2	3	4
1	0	1	0	1
2	1	0	1	0
3	0	0	0	1
4	0	0	1	0

==> XMT

c. Nova opção.

{█ _ _ }

XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

***OPCAO LIDA = 10

MOSTRA NA TELA A PROXIMA LISTA DE OPCOES

A PROXIMA LISTA SERA A SEGUNDA

PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA

MESMO QUE NAO CONSTE DA LISTA APRESENTADA

==>XMT

Informações solicitadas.

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Apresentará a segunda lista de opções, somente se o grafo já foi editado, uma vez que nas listas seguintes não existe nenhuma opção de edição.

b. Disposição da tela.

Pode-se escolher qualquer opção, mesmo que não conste da lista apresentada; para tanto é necessário saber qual o número da opção desejada.

{█_}

XMT{_}

***ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 02):

- 11 - VALORAR UM GRAFO
- 12 - CAMINHO MINIMO (FLOYD)
- 13 - DETERMINA A EXISTENCIA DE CIRCUITO (DEMOUCRON)
- 14 - CAMINHO MAXIMO (FLOYD)
- 15 - RAI0, DIAMETRO (GRAFOS NAO VALORADOS)
- 16 - COMPONENTES F-CONEXAS (MALGRANGE)
- 17 - RAI0, DIAMETRO (GRAFO VALORADOS)
- 18 - FLUXO MAXIMO (FORD-FULKERSON)
- 19 - CAMINHO MINIMO (DIJKSTRA)
- 20 - PROXIMA LISTA DE OPCOES

***OPCAO LIDA = 11

PERMITE VALORAR UM GRAFO

PODE-SE DAR UM VALOR CONSTANTE PARA TODOS ARCOS/ARESTAS

GRAFOS NAO ORIENTADOS TERA O VALORES SIMETRIZADOS

PODE-SE PEDIR UMA SAIDA DA MATRIZ DE VALORES

A IMPRESSAO CONTEM DUAS CASAS DECIMAIS

O MAXIMO VALOR DEVERA SER MENOR QUE 10 ELEVADO A 5

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{■_ _}

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 11, somente se o grafo já foi editado.

b. Entrada do valor.

{█ _ _ }	XMT{ _ }
***DESEJA DAR UM VALOR CONSTANTE A TODOS ARCOS/ARESTAS? SIM OU NAO	

SIM - Executa o item c para entrada da constante.

NAO - Executa o item d para entrada de valores diferentes para todos os arcos/arestas.

c. Valor constante.

{█ _ _ _ _ }	XMT{ _ }
***ENTRE COM O VALOR DA CONSTANTE	

Seja c o valor da constante.

São permitidos valores no intervalo:

$$0 < c \leq 99999.$$

A precisão para a saída será sempre de duas casas decimais. Se na entrada o número tiver mais de duas casas, internamente ele será processado com a precisão indicada, porém a impressão será truncada.

O número pode ser digitado em qualquer posição do campo, desde que contenha o ponto decimal. Caso não se deseje colocar o ponto, o número deve ser enquadrado à direita e o valor interno conterá duas casas decimais, contando da direita para esquerda.

Entradas válidas:		Valor interno
{035.22}	XMT{■}	35.22
{ 45.3 }	XMT{■}	45.30
{78. }	XMT{■}	78.00
{ 3672}	XMT{■}	36.72
{999999}	XMT{■}	9999.99
{99999.}	XMT{■}	99999.00

Em seguida será dada uma mensagem informando que o grafo já foi valorado.

Caso se tenha optado pela execução do item c , o item d seguinte não será executado.

d. Valores diferentes.

{ ■ _ _ _ _ } { _ _ _ _ } { _ _ _ _ } { _ _ _ _ }

XMT { _ }

***ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(u₁) (u₂) (u₃)(u₉).

Onde $u_1, u_2, u_3, \dots, u_9$ indicam os arcos/arestas do grafo que devem ser entrados naquele instante. Os arcos/arestas u são indicados pelos seus vértices inicial e final.

São solicitados valores de nove arcos/arestas de cada vez partindo-se dos arcos iniciados no vértice 1, depois dos iniciados no vértice 2, e assim por diante.

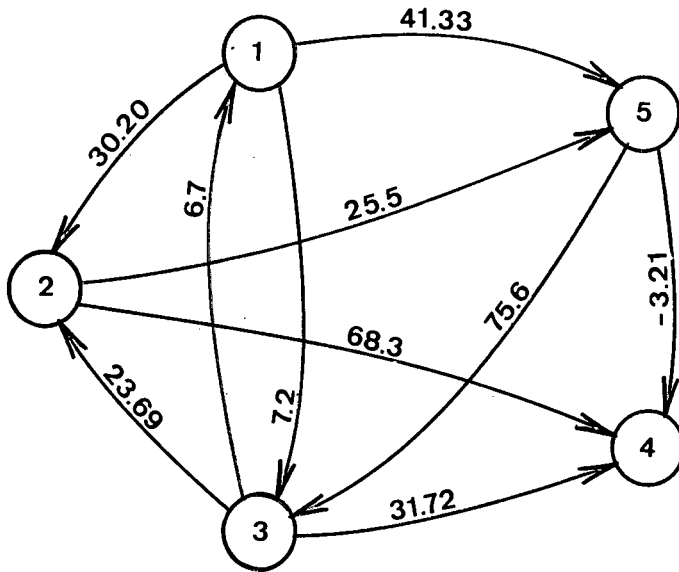
Para os grafos não orientados s̄o serão solicitados os valores dos arcos oriundos do triângulo inferior da matriz de adjacência, porém os valores serão simetrizados para os demais arcos.

S̄o são solicitados os valores para os arcos/arestas realmente existentes, definidos pela matriz de adjacência.

Se v é o valor, s̄o serão permitidos valores no intervalo:

$$- 9999. \leq v \leq 99999.$$

Exemplo:



Primeira tela:

{30.20 } {7.2 } { 41.33}{68.3 } { 25.5}{6.7 } {23.69 } { 31.72}{75.6 }

XMT {■}

***ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(1 - 2)(1 - 3)(1 - 5)(2 - 4)(2 - 5)(3 - 1)(3 - 2)(3 - 4)(5 - 3)

Segunda tela:

{- 3.21}

XMT{■}

***ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(5 - 4)

Em seguida será dada uma mensagem informando que o grafo já foi valorado.

e. Outra valoração.

{■ _ _ }

XMT{ _ }

***PREFERE UMA VALORAÇÃO DIFERENTE? SIM OU NAO

SIM - Reiniciará a rotina a partir do item b, para uma nova valoração. Os valores anteriormente dados serão destruídos.

NAO - Segue a execução.

f. Matriz de valores.

{■ _ _ }

XMT{ _ }

***DESEJA IMPRIMIR A MATRIZ DE VALORES? SIM OU NAO

SIM - Emitirá uma listagem da matriz de valores do grafo. O formato de impressão conterá no máximo 10 colunas, no caso de grafos que possuam mais de 10 vértices, cada vértice virá impresso em duas ou mais linhas. Os valores infinitos, isto é, vértices não ligados, terão o campo impresso com asteriscos (*). Será dada uma mensagem de que a matriz já foi impressa, o escaninho da listagem será o indicado pelo usuário ou o de número igual ao do terminal em uso.

NAO - Segue a execução.

g. Nova opção

{ ■ _ _ }	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

***OPCAO LIDA = 12

ALGORITMO DE FLOYD PARA CAMINHO MINIMO

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

CALCULA A MATRIZ DE DISTANCIA MINIMA DO GRAFO

CALCULA A MATRIZ DE ROTEAMENTO DO GRAFO

PODE-SE IMPRIMIR A MATRIZ DE DISTANCIA E DE ROTEAMENTO

PODE-SE PERGUNTAR A DISTANCIA MINIMA ENTRE DOIS VERTICES

OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 12, somente se o grafo já foi editado. Esta rotina só é permitida para grafos já valorados; caso o grafo ainda não tenha recebido valores a rotina de valoração (opção 11) será chamada automaticamente e dará uma mensagem de advertência, seguindo a execução.

b. Matriz de distância.

{ ■ _ _ }	XMT { _ }
***DESEJA IMPRIMIR A MATRIZ DE DISTANCIA? SIM OU NAO	

SIM - Emitirá uma listagem da matriz de distância mínima do grafo. O formato de impressão conterá no máximo 10 colunas, no caso de grafos que possuam mais de 10 vértices, cada vértice virá impresso em duas ou mais linhas. Os valores infinitos, isto é, vértices entre os quais não existe nenhum caminho, terão o campo impresso com asteriscos(*). Será dada uma mensagem de que a matriz foi impressa, o escaninho da listagem será o indicado pelo usuário ou o de número igual ao do terminal em uso.

NAO - Segue a execução.

c. Matriz de roteamento.

{ ■ _ _ }	XMT { _ }
***DESEJA IMPRIMIR A MATRIZ DE ROTEAMENTO? SIM OU NAO	

SIM - Emitirá uma listagem da matriz de roteamento mínimo do grafo.

NAO - Segue a execução.

d. Caminho m̄nimo

```
{█_} {__} XMT{__}
***INDIQUE ENTRE QUE VERTICES I - J DESEJA O CAMINHO MINIMO(MAXIMO)
```

Seja n o n̄mero de v̄rtices do grafo.

S̄o ser̄o aceitos valores de i e j tais que:

$$0 < i \leq n \quad \text{e} \quad 0 < j \leq n$$

Enquadrar o n̄mero ̄ direita, precedido de branco ou zero.

Entradas v̄lidas:

```
{01}{15} XMT{█}
{ 3}{ 9} XMT{█}
```

e. Disposiç̄o da tela.

Ap̄s escolhidos os dois v̄rtices entre os quais se deseja saber o caminho m̄nimo, ser̄a informado o valor deste caminho e os v̄rtices pertencentes a ele. Caso n̄o exista nenhum caminho entre os v̄rtices indicados ser̄a emitida uma mensagem.

Exemplos:

Se $i = 1$ e $j = 15$

```
***VERTICES DO CAMINHO MINIMO ENTRE ( 1 E 15)
  1  5  7  9 12 15
***DISTANCIA MINIMA ENTRE OS VERTICES ( 1 E 15) 358.32
==>XMT
```

Se $i = 3$ e $j = 9$

***VERTICES DO CAMINHO MINIMO ENTRE (3 E 9)

3 9

***DISTANCIA MINIMA ENTRE OS VERTICES (3 E 9) 120.31

== > XMT

Se $i = 5$ e $j = 30$

***NAO EXISTE NENHUM CAMINHO ENTRE OS VERTICES (5 E 30)

== > XMT

f. Outro caminho

{ ■ _ _ }

XMT{ _ }

***DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES? SIM OU NAO

SIM - Retornará ao item d para que se indiquem os novos vértices entre os quais se deseja saber o caminho mínimo.

NAO - Segue a execução.

g. Nova opção.

{■ _ _}	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

h. Observação geral.

O algoritmo de Floyd utilizado admite a entrada de valores negativos para os arcos, porém neste caso deve-se estar certo de que o grafo não contenha ciclos negativos, pois as saídas podem fornecer resultados inesperados.

***OPCAO LIDA = 13

ALGORITMO DE DEMOUCRON

DETERMINA OS NIVEIS DO GRAFO

INDICA A EXISTENCIA DE CIRCUITO

INDICA OS NIVEIS EXISTENTES ATE ENCONTRAR

O PRIMEIRO CIRCUITO

GRAFOS NAO ORIENTADOS, INDICA A EXISTENCIA DE CICLO.

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{ _ _ }

XMT{ _ }

***DESEJA SABER OS NIVEIS EXISTENTES NO GRAFO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 13, somente se o grafo já foi editado.

b. Grafo sem circuito.

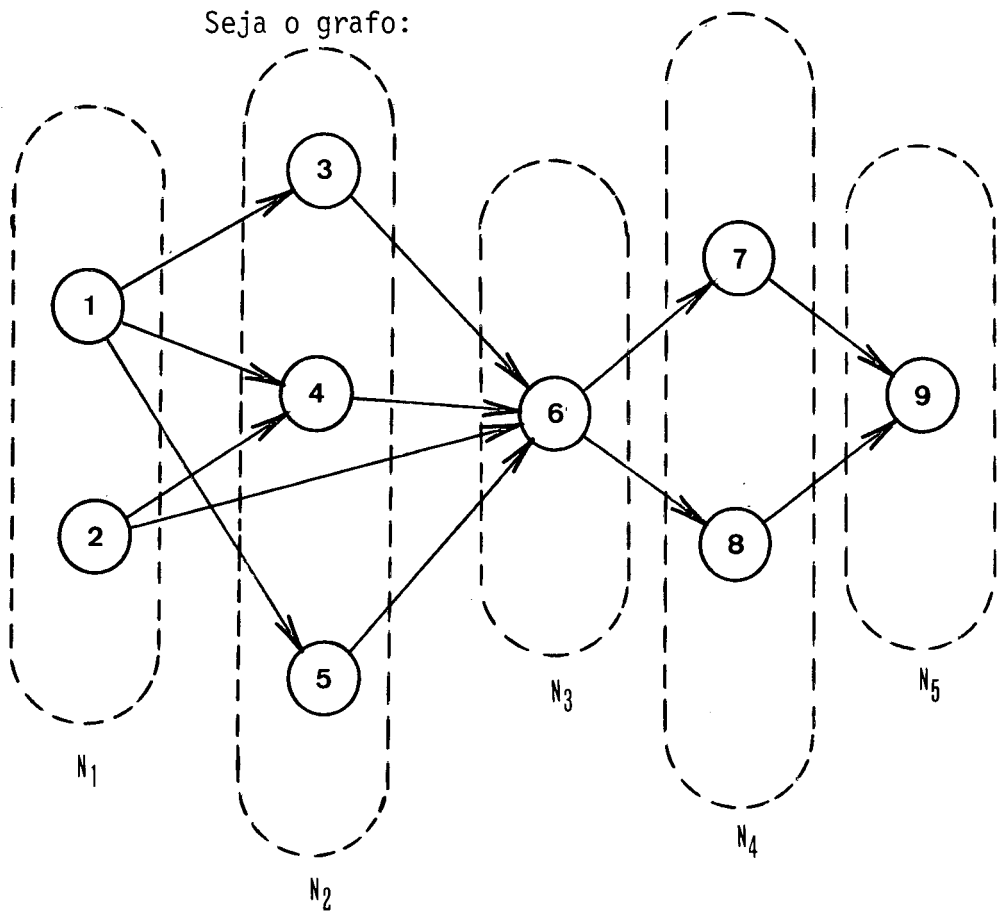
Se o grafo não possui circuito será feita a seguinte solicitação.

<input type="checkbox"/>	XMT{ }
***DESEJA SABER OS NIVEIS EXISTENTES NO GRAFO? SIM OU NAO	

SIM - Serão apresentados os níveis existentes e os vértices pertencentes a cada nível.

Exemplo:

Seja o grafo:



Disposição da tela.

```
***NIVEIS EXISTENTES NO GRAFO

NIVEL 1 ... 1 2
NIVEL 2 ... 3 4 5
NIVEL 3 ... 6
NIVEL 4 ... 7 8
NIVEL 5 ... 9

==>XMT
```

Serão apresentados até no máximo 10 níveis por tela.

Segue a execução no item d.

NAO - Segue a execução no item d.

c. Grafo com circuito.

Se o grafo apresenta algum circuito, dois tipos de mensagem podem ser dados:

- (a) Foi possível alguma partição em níveis antes de ser encontrado o primeiro circuito. Neste caso serão apresentados os níveis existentes até este ponto.

Exemplo: Se no grafo anterior o vértice 9 fosse ligado ao



***SEU GRAFO APRESENTA PELO MENOS UM CIRCUITO OU CICLO

***OS NIVEIS EXISTENTES ATE O MOMENTO SAO:

NIVEL 1 ... 1 2

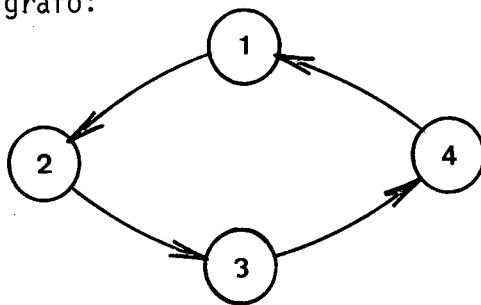
NIVEL 2 ... 3 4 5

==>XMT

(b) Não foi possível fazer nenhuma partição. Neste caso a mensagem será apenas a de existencia do circuito.

Exemplo:

Seja o grafo:



Disposição da tela.

***SEU GRAFO APRESENTA PELO MENOS UM CIRCUITO OU CICLO

==>XMT

d. Nova opção.

{ _ _ }

XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

e. Grafos não orientados.

Neste caso a rotina executarã apenas os itens a e d, porẽm serã dadas as mensagens sobre a existẽncia ou não de ciclo.

***OPCAO LIDA = 14

ALGORITMO DE FLOYD PARA CAMINHO MAXIMO

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

O GRAFO NAO PODE TER NENHUM CIRCUITO

CALCULA O CAMINHO CRITICO NUMA REDE PERT

CALCULA A MATRIZ DE DISTANCIA MAXIMA DO GRAFO

PODE-SE IMPRIMIR QUALQUER DAS MATRIZES

PODE-SE PEDIR A DISTANCIA MAXIMA ENTRE DOIS VERTICES

OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{█ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 14, somente se o grafo já foi editado. Esta rotina sô é permitida para grafos já valorados; se o grafo ainda não recebeu valores, a rotina de valoração (opção 11) serã chamada automaticamente e darã uma mensagem de advertência, seguindo a execução.

b. Níveis do grafo.

A opção 13 serã chamada automaticamente para verificar se o grafo possui ou não algum circuito. No caso afirmativo serã dada uma mensagem de advertência, (pois este algoritmo sô é vãlido para grafos sem circuito), desviando apôs para o item h, para a escolha de nova opção.

Não havendo nenhum circuito, serã então indicados os níveis existentes, se isto for solicitado, seguindo a execução.

c. Matriz de distância

{■ _ _ }

XMT{ _ }

***DESEJA IMPRIMIR A MATRIZ DE DISTANCIA? SIM OU NAO

SIM - Emitirã uma listagem da matriz de distâncias mãximas do grafo. O formato de impressã conterã no mãximo 10 colunas, no caso de grafos que possuam mais de 10 vãrtices, cada vãrtice virã impresso em duas ou mais linhas. Os valores infinitos, isto é, vãrtices entre os quais não existe nenhum caminho, terã o campo

impresso com asteriscos (*). Serã dada uma mensagem de que a matriz foi impressa, o escaninho da listagem serã o indicado pelo usuãrio ou o de nũmero igual ao do terminal em uso.

NAO - Segue a execuãõ

d. Matriz de roteamento

```
{█ _ } XMT{ _ }  
***DESEJA IMPRIMIR A MATRIZ DE ROTEAMENTO? SIM OU NAO
```

SIM - Emitirã uma listagem da matriz de roteamento mãximo do grafo.

NAO - Segue a execuãõ.

e. Caminho mãximo

```
{█ _ } { _ } XMT{ _ }  
***INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO(MAXIMO)
```

Seja n o nũmero de vẽrtices do grafo.

Sõ serão aceitos valores de i e j tais que:

$$0 < i \leq n \quad \text{e} \quad 0 < j \leq n$$

Enquadrar o nũmero ã direita, precedido de branco ou zero.

Entradas vãlidas:

{01}{19} XMT{█ }

{12}{02} XMT{█ }

f. Disposição da tela:

Após escolhidos os dois vértices entre os quais se deseja saber o caminho máximo, será informado o valor deste caminho e os vértices pertencentes a ele. Caso não exista nenhum caminho entre os vértices indicados será emitida uma mensagem:

Exemplos:

Se $i = 2$ e $j = 18$

```
***VERTICES DO CAMINHO MAXIMO ENTRE ( 2 E 18 )
  2   3  14  16  18
***DISTANCIA MAXIMA ENTRE OS VERTICES ( 2 E 18 )    203.01
==>XMT
```

Se $i = 15$ e $j = 5$

```
***VERTICES DO CAMINHO MAXIMO ENTRE ( 15 E 5 )
 15  10  3  2  5
***DISTANCIA MAXIMA ENTRE OS VERTICES ( 15 E 5 ) 152.00
==>XMT
```

Se $i = 21$, $j = 37$ e $n = 35$ (número de vértices)

```
***VALORES LIDOS I = 21 J = 37 INCOMPATIVEL COM NUMERO DE VERTICES
TENTE NOVAMENTE

==>XMT
```

Neste caso a rotina retorna ao item e para uma nova entrada de um par de vértices.

Se $i = 5$ e $j = 16$

```
***NAO EXISTE NENHUM CAMINHO ENTRE OS VERTICES ( 5 E 16 )

==>XMT
```

g. Outro caminho.

```
{ ■ _ _ } XMT{ _ }

***DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES? SIM OU NAO
```

SIM - Retornará ao item e para que se indiquem os novos vértices entre os quais se deseja saber o caminho máximo.

NAO - Segue a execução.

h. Nova opção.

```
{ ■ _ _ } XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO
```

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

***OPCAO LIDA = 15

RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAMENTOS

VALIDA PARA GRAFOS NAO VALORADOS

CALCULA O RAO E O DIAMETRO INTERIORES E EXTERIORES

DETERMINA O CENTRO E VERTICES PERIFERICOS

DETERMINA OS AFASTAMENTOS DOS VERTICES

CALCULA A MATRIZ DE DISTANCIAS

PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS

VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{ __ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 15, somente se o grafo já foi editado.

b. Matriz de distância e afastamentos.

Como esta rotina é usada para grafos não valorados todos os dados são obtidos a partir da matriz de adjacência na forma booleana, portanto se o grafo já foi valorado estes valores não são levados em consideração.

```
{█ _ _ }
```

```
XMT{ _ }
```

```
***DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E AFASTAMENTOS? SIM OU NAO
```

SIM - Emitirá a listagem da matriz de distância e os valores dos afastamentos para todos os vértices. Os valores in finitos, isto é, vértices entre os quais não existe nenhum caminho, terão o campo impresso com asteriscos (*) na matriz de distâncias. Os afastamentos infinitos também serão indicados com asteriscos. Será dada uma mensagem de que os pedidos já foram impressos. O escaninho da listagem será o indicado pelo usuário ou o de número igual ao do terminal em uso.

NAO - Segue a execução.

c. Disposição da tela.

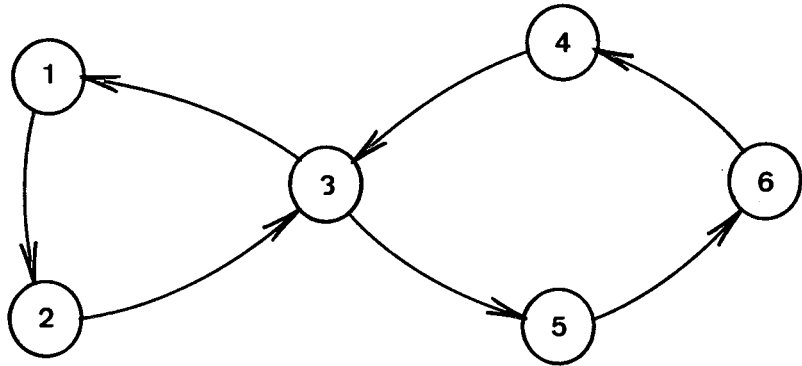
Ao final são dadas informações concernentes à opção: raio, diâmetro, centros e periféricos. Indicarão também, se existirem, os centros bi-direcionais.

Caso ocorra a ausência de centros, isto será informado.

Os valores infinitos terão os campos impressos com asteriscos (*).

Exemplos:

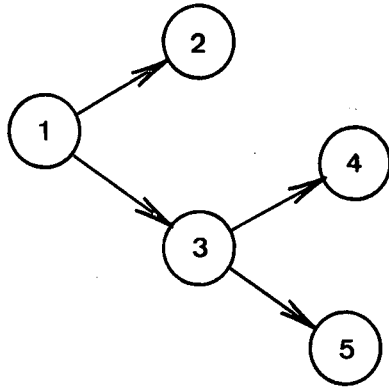
(i) Seja o grafo:



```
***INFORMACOES SOBRE O GRAFO  
  
RAIO EXTERIOR 3  
DIAMETRO EXTERIOR 5  
CENTROS EXTERIORES 3 4  
PERIFERICOS EXTERIORES 1 5  
  
RAIO INTERIOR 3  
DIAMETRO INTERIOR 5  
CENTROS INTERIORES 3 5  
PERIFERICOS INTERIORES 2 4  
CENTROS BI-DIRECIONAIS 3  
  
==>XMT
```

Seja o grafo

(ii)



***INFORMACOES SOBRE O GRAFO

RAIO EXTERIOR 2

DIAMETRO EXTERIOR ****

CENTROS EXTERIORES 1

PERIFERICOS EXTERIORES 2 3 4 5

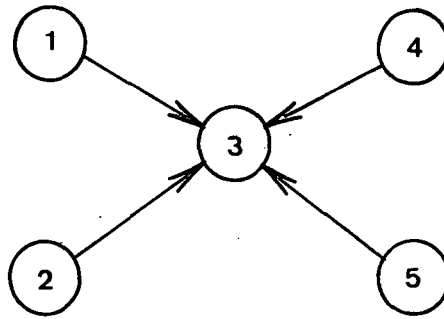
RAIO INTERIOR ****

DIAMETRO INTERIOR ****

***AUSENCIA DE CENTROS INTERIORES

==>XMT

(iii) Seja o grafo



```
***INFORMACOES SOBRE O GRAFO
RAIO EXTERIOR ****
DIAMETRO EXTERIOR ****
***AUSENCIA DE CENTROS EXTERIORES

RAIO INTERIOR 1
DIAMETRO INTERIOR ****
CENTROS INTERIORES 3
PERIFERICOS INTERIORES 1 2 4 5

==>XMT
```

d. Nova opção

```
{█ _ _} XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO
```

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçãõ.

***OPCAO LIDA = 16

DESCOMPOSICAO EM COMPONENTES F-CONEXAS (MALGRANGE)

DETERMINA AS COMPONENTES F-CONEXAS

INDICA SE O GRAFO E F-CONEXO

UTILIZA O ALGORITMO DE MALGRANGE.

==>XMT

Informações solicitadas.

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execuãõ da opção 16, somente se o grafo
já foi editado.

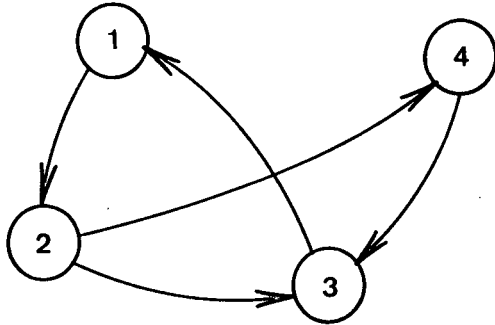
b. Disposiãõ da tela.

Apresentarã a partiãõ do grafo em suas componentes
fortemente conexas, indicando os vèrtices pertencentes a cada um.

Se o grafo for f-conexo isto serã indicado.

Exemplos:

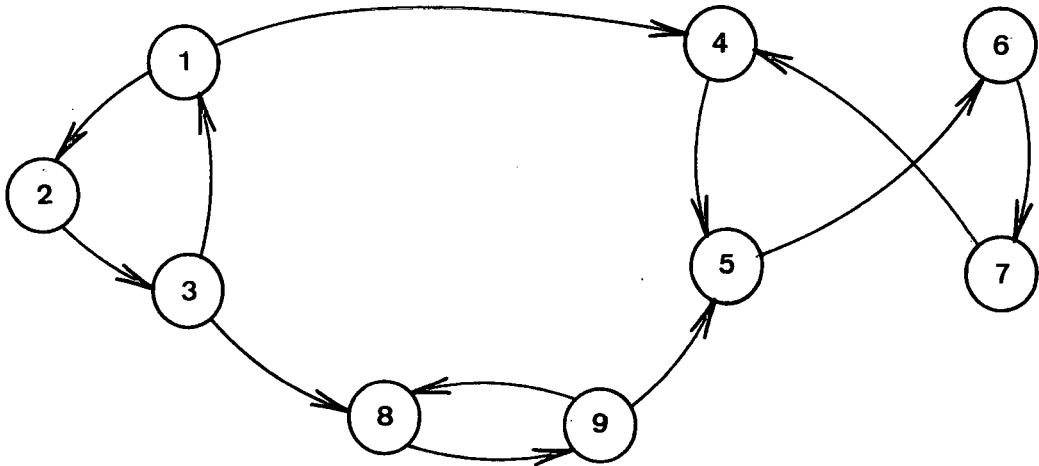
(i) Seja o grafo:



***O SEU GRAFO E FORTEMENTE CONEXO

==>XMT

(ii) Seja o grafo



***COMPONENTES F-CONEXAS DO GRAFO

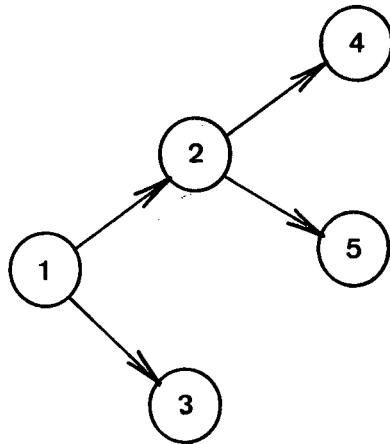
VERTICES ... 1 2 3

VERTICES ... 4 5 6 7

VERTICES ... 8 9

==>XMT

(iii) Seja o grafo



***COMPONENTES F-CONEXAS DO GRAFO

VERTICE ... 1

VERTICE ... 2

VERTICE ... 3

VERTICE ... 4

VERTICE ... 5

==>XMT

c. Nova opção.

{ _ _ }

XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

***OPCAO LIDA = 17

RAIO, DIAMETRO,CENTROS,PERIFERICOS E AFASTAMENTOS

VALIDA PARA GRAFOS VALORADOS

CALCULA O RAI0 E O DIAMETRO INTERIORES E EXTERIORES

DETERMINA OS CENTROS E VERTICES PERIFERICOS

CALCULA A MATRIZ DE DISTANCIAS

PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS

VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

PODE-SE DAR PESO AOS VERTICES

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 17, somente se o grafo já foi editado. Esta rotina sô é aplicãvel a grafos já valorados; caso o grafo ainda não tenha recebido valores, a rotina de valoraçã0 (opçã0 11) serã chamada automaticamente e darã uma mensagem de advertência seguindo a execução.

b. Pesos aos v̄rtices.

```
{█ _ _ }           XMT{ _ }  
  
***DESEJA DAR VALORES AOS VERTICES? SIM OU NAO
```

SIM - Executa o item c para a entrada dos valores.

NAO - Executa o item d. Neste caso os pesos dos v̄rtices s̄o assumidos como tendo valor unit̄rio.

c. Entrada dos valores.

A tela ser̄ formatada para a entrada dos valores relativos a dez v̄rtices de cada vez. A solicitaç̄o ser̄ feita para os v̄rtices na ordem crescente a partir de um.

```
{█ _ _ _ } { _ _ _ } { _ _ _ } ..... { _ _ _ }  
  
XMT{ _ }  
  
***ENTRE COM OS VALORES DO VERTICE  $i_1$  ATE O VERTICE  $i_2$ 
```

Onde i_1 indica o v̄rtice a partir do qual deve ser fornecido o valor do peso e i_2 indica o v̄rtice final para a entrada em quest̄o.

Se p ̄ o peso, s̄o ser̄o permitidos valores no intervalo:

$$-999. \leq p \leq 9999.$$

O número pode ser digitado em qualquer posição do campo, desde que contenha o ponto decimal. Caso não se deseje colocar o ponto o número deve ser enquadrado à direita e o valor interno conterá duas casas decimais, contando da direita para esquerda.

Entradas válidas.

Supor um grafo de cinco vértices:

```
{.32 }{0.46 }{3.65 }{4.   }{ 672}
```

```
XMT{█}
```

d. Matriz de distância e afastamentos.

```
{█ _ _ }
```

```
XMT{ _ }
```

```
***DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E AFASTAMENTOS? SIM OU NAO
```

SIM - Emitirá a listagem da matriz de distância e os valores dos afastamentos para todos os vértices. Os valores infinitos, (relativos a pares de vértices entre os quais não existe nenhum caminho) terão o campo impresso com asteriscos (*) na matriz de distância. Os afastamentos infinitos também serão impressos com asteriscos. O formato de impressão da matriz conterá no máximo 10 colunas; no caso de grafos que possuam mais de 10 vértices, cada vértice virá impresso em duas ou mais linhas.

Será dada uma mensagem de que os pedidos já foram impressos, o escaninho da listagem será o indicado pelo usuário ou o de número igual ao do terminal em uso.

NAO - Segue a execução.

e. Disposição da tela.

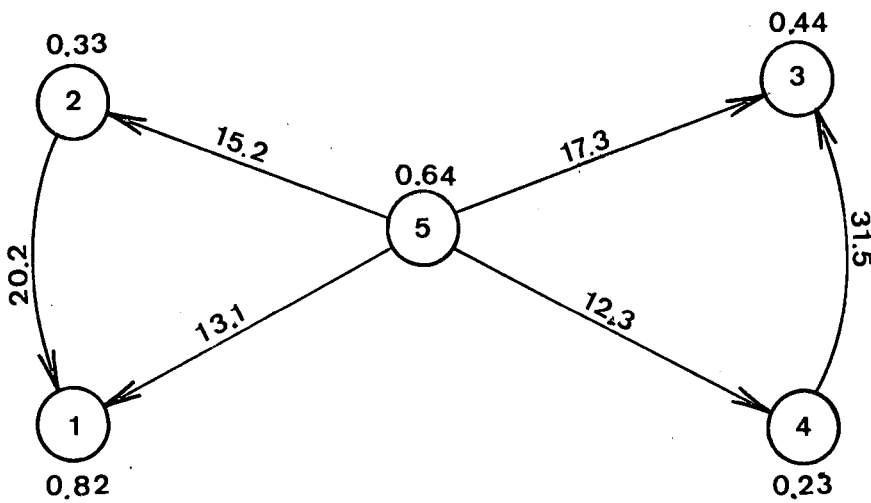
Ao final são dadas as informações concernentes à opção: Raio, diâmetro, centros e periféricos. Indicarão também, se existirem, os centros bi-direcionais.

Em caso de ausência de centros, isto será informado.

Os valores infinitos terão os campos impressos com asteriscos (*).

Exemplo:

Seja o grafo abaixo, onde os números sobre os arcos indicam as capacidades dos mesmos, e os valores próximos aos vértices indicam os pesos atribuídos a cada um deles.



```
***INFORMACOES SOBRE O GRAFO
RAIO EXTERIOR      10.74
DIAMETRO EXTERIOR *****
CENTROS EXTERIORES  5
PERIFERICOS EXTERIORES  1  2  3  4

RAIO INTERIOR *****
DIAMETRO INTERIOR *****
***AUSENCIA DE CENTROS INTERIORES
==>XMT
```

f. Nova opção.

```
{■ _ _}          XMT{ _ }

***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO
```

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

```
***OPCAO LIDA = 18
```

```
ALGORITMO DE FORD-FULKERSON (FLUXO MAXIMO)
```

```
VALIDA PARA GRAFOS VALORADOS
```

```
INDICA O VALOR DO FLUXO MAXIMO
```

```
IMPRIME OS ARCOS PARA O FLUXO MAXIMO
```

```
SAO INDICADOS OS ARCOS SATURADOS
```

```
O FLUXO E DETERMINADO ENTRE O VERTICE 1 e NVER
```

```
==>XMT
```

Informações solicitadas:

a. Escolher outra opção.

```
{█ _ _ }
```

```
XMT{ _ }
```

```
***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO
```

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 18, somente se o grafo já foi editado. Esta rotina sã é permitida para grafos já valorados; caso o grafo ainda não tenha recebido valores a rotina de valoração (opção 11) serã chamada automaticamente e darã uma mensagem de advertência. Ao findar a execução, serã apresentada na tela uma mensagem indicando o valor do fluxo m̃ximo, entre o primeiro e ũltimo ṽrtices.

No caso, por exemplo de um grafo de dez v̄ertices, o v̄ertice um ser̄a a fonte e o v̄ertice dez o sumidouro.

O valor do fluxo m̄aximo ser̄a fornecido com duas casas decimais.

b. Valores dos arcos.

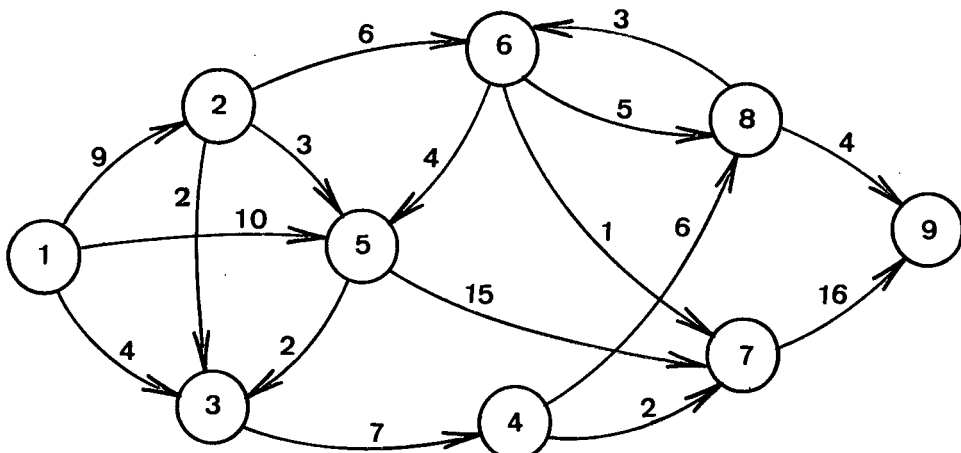
```
{█ _ _ } XMT{ _ }  
  
***DESEJA IMPRIMIR OS VALORES DOS ARCOS  
PARA O FLUXO MAXIMO? SIM OU NAO
```

SIM - Imprimir̄a os valores dos fluxos de cada um dos arcos, correspondentes ao fluxo m̄aximo encontrado. Os arcos saturados (capacidade igual ao fluxo parcial) ser̄a indicados. Um exemplo da sāida pode ser visto no item seguinte.

NAO - Segue a execūc̄o no item d.

c. Disposīc̄o da tela

Seja o seguinte grafo, onde os valores sobre os arcos indicam as capacidades dos mesmos:



Neste caso a tela assumira a seguinte disposiç~ao:

```
***O FLUXO MAXIMO OBTIDO POR FORD-FULKERSON FOI  
  
20.00  
  
==>XMT
```

Se for solicitada a impress~ao dos valores dos arcos como indicado no item b, a listagem conter~a a seguinte forma:

VALORES DOS ARCOS PARA O FLUXO MAXIMO (FORD-FULKERSON)

ARCO	1	2	=	6.00	
ARCO	1	3	=	4.00	SATURADO
ARCO	1	5	=	10.00	SATURADO
ARCO	2	3	=	2.00	SATURADO
ARCO	2	5	=	3.00	SATURADO
ARCO	2	6	=	1.00	
ARCO	3	4	=	6.00	
ARCO	4	7	=	2.00	SATURADO
ARCO	4	8	=	4.00	
ARCO	5	7	=	13.00	
ARCO	6	7	=	1.00	SATURADO
ARCO	7	9	=	16.00	SATURADO
ARCO	8	9	=	4.00	SATURADO

Na tela será dada uma informação de que os dados foram impressos; o escaninho da listagem será o indicado pelo usuário ou o de número igual ao do terminal em uso.

d. Nova opção.

{█ _ _ }	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.

***OPCAO LIDA = 19

ALGORITMO DE DIJKSTRA (CAMINHO MINIMO)

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

DEVE-SE INDICAR O VERTICE RAIZ

FORNECE A DISTANCIA DA RAIZ A TODOS OS VERTICES

PODE-SE ESCOLHER QUALQUER VERTICE PARA RAIZ

USAR FLOYD PARA SABER OS VERTICES DO CAMINHO

==>XMT

Informações solicitadas:

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Continuará a execução da opção 19, somente se o grafo já foi editado. Esta rotina só é permitida para grafos já valorados; se o grafo ainda não recebeu valores, a rotina de valoração (opção 11) será chamada automaticamente e dará uma mensagem de advertência, seguindo a execução.

b. Entrada da raiz.

```
{█ _}          XMT{█ }  
  
***ENTRE COM O VALOR DA RAIZ DESEJADA
```

Sendo r a raiz e n o número de vértices do grafo, só serão aceitos valores tais que:

$$0 < r \leq n$$

Enquadrar o número à direita precedido de branco ou zero.

Entradas válidas

{15} XMT{█ }

{ 1} XMT{█ }

{05} XMT{█ }

c. Disposição da tela.

Após a execução do programa, serão apresentadas na tela as distâncias da raiz indicada a cada um dos vértices do grafo, além de uma mensagem sugerindo a chamada da opção 12 (Floyd para caminho mínimo), caso se deseje saber os vértices pertencentes aos caminhos.

Para grafos com mais de 15 vértices, são apresentadas na tela as distâncias da raiz para grupos de 15 vértices de cada vez.

Os valores infinitos, ou seja os vértices para os quais não existe nenhum caminho partindo da raiz, terão o campo impresso com asteriscos(*).

Usando-se como exemplo o grafo apresentado na opção 18 e escolhendo-se como raiz o vértice cinco, a tela terá a seguinte disposição:

```
***DISTANCIAS ENTRE A RAIZ 5 E OS VERTICES:

VERTICE 1 ... *****
VERTICE 2 ... *****
VERTICE 3 ...      2.00
VERTICE 4 ...      9.00
VERTICE 5 ...      0.00
VERTICE 6 ...     18.00
VERTICE 7 ...     11.00
VERTICE 8 ...     15.00
VERTICE 9 ...     19.00

***PARA SABER OS VERTICES DO CAMINHO USAR FLOYD (OPCAO = 12)

==>XMT
```

d. Outra raiz.

```
{█ _ _}          XMT{ _ }

***DESEJA MUDAR O VALOR DA RAIZ? SIM OU NAO
```

SIM - Retornará ao item b para entrada de uma nova raiz

NAO - Segue a execução

e. Nova opção.

{█ _ _}	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalização.

***OPCAO LIDA = 20

MOSTRA NA TELA A PROXIMA LISTA DE OPCOES

A PROXIMA LISTA SERA A TERCEIRA

PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA

MESMO QUE NAO CONSTE DA LISTA APRESENTADA

DEVE ESCOLHER UMA OPCAO NA PROXIMA LISTA POIS SERA A ULTIMA

==>XMT

Informações solicitadas.

a. Escolher outra opção.

{█ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Apresentarã a terceira lista de opções, somente se o grafo já foi editado, uma vez que na lista seguinte não existe nenhuma opção de edição.

b. Disposição da tela.

{█}_	XMT{ }
***ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 03):	
21	- ARVORE PARCIAL MINIMA (KRUSKAL)
22	- TORNAR UM GRAFO, NAO ORIENTADO
23	- LIVRE
24	- LIVRE
25	- LIVRE
26	- LIVRE
27	- LIVRE
28	- LIVRE
29	- LIVRE
30	- VOLTA A LISTA 01 (ESTA E A ULTIMA LISTA DE OPCOES)

Pode ser escolhida qualquer opção, mesmo que não conste da lista apresentada; para tanto é necessário que se saiba o número da opção desejada.

Como esta é a última lista existem nela algumas opções livres, que devem ser usadas para ampliações futuras. Caso seja escolhida alguma destas opções ocorrerá o encerramento de execução do programa, dando as mensagens de finalização.

***OPCAO LIDA = 21

ALGORITMO DE KRUSKAL (ARVORE PARCIAL MINIMA)

VALIDA SO PARA GRAFOS NAO ORIENTADOS E VALORADOS

FORNECE OS ARCOS DA ARVORE PARCIAL

==>XMT

Informações solicitadas.

a. Escolher outra opção.

{■ _ _ }

XMT{ _ }

***PREFERE ESCOLHER OUTRA OPCAO? SIM OU NAO

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execução da opção 21, somente se o grafo já foi editado. Como esta rotina sã é permitida para grafos não orientados e já valorados, caso estas condições não sejam satisfeitas, as rotinas de eliminação da orientação (opção 22) e de valoração (opção 11) serão chamadas automaticamente, dando uma mensagem de advertência.

b. Arcos da árvore parcial.

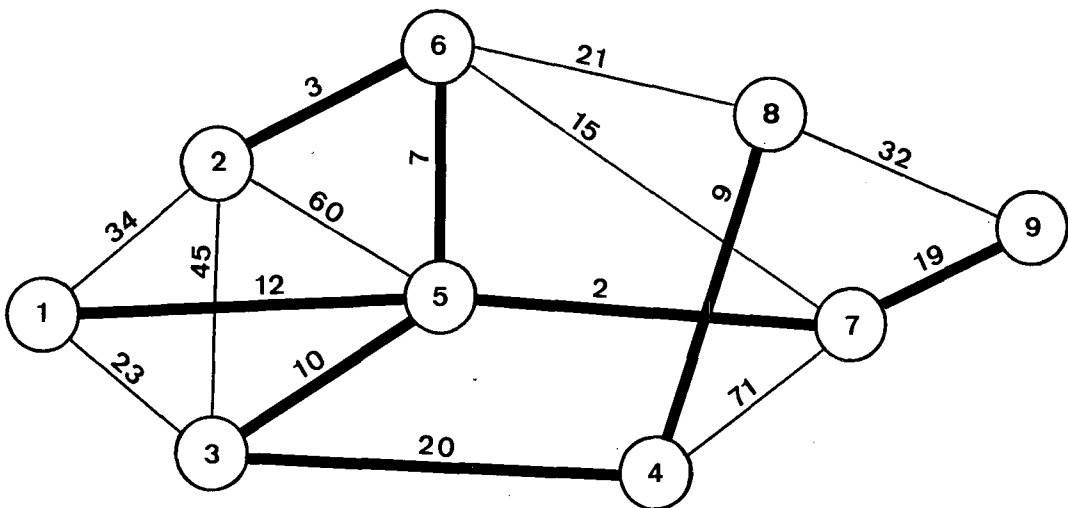
<input type="checkbox"/>	XMT{ }
***DESEJA SABER OS ARCOS DA ARVORE PARCIAL MINIMA? SIM OU NAO	

SIM - Apresentará na tela os arcos que constituem a árvore parcial mínima, juntamente com os valores associados a cada um. Ver exemplo no item c.

NAO - Segue a execução no item d.

c. Disposição da tela.

Como exemplo, seja o grafo abaixo, onde os valores sobre os arcos indicam os custos dos mesmos.



A tela assumirá a seguinte disposição:

***ARCOS DA ARVORE PARCIAL MINIMA (KRUSKAL)			
ARCO	5	- 7	2.00
ARCO	2	- 6	3.00
ARCO	5	- 6	7.00
ARCO	4	- 8	9.00
ARCO	3	- 5	10.00
ARCO	1	- 5	12.00
ARCO	7	- 9	19.00
ARCO	3	- 4	20.00

==>XMT

A árvore obtida no exemplo acima está indicada pelos traços mais fortes na figura.

Se, para algum grafo, forem gerados mais de 15 arcos para a árvore, serão apresentados na tela grupos de 15 arcos de cada vez.

d. Nova opção

<input type="checkbox"/> _ _	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrará a primeira lista de opções para escolha.

NAO - Dará as mensagens de finalização.


```
***OPCAO LIDA = 22

      DESORIENTAR UM GRAFO

ALTERA O GRAFO ORIGINAL

O GRAFO PASSA A SER NAO ORIENTADO

SE HOUVER O ARCO (I,J) HAVERA O ARCO (J,I)

NOS GRAFOS VALORADOS VAL(I,J) = VAL (J,I)

PARA VALORES NOS DOIS SENTIDOS TEM-SE:

      VAL(I,J) = MAX (VAL(I,J), VAL(J,I))

==>XMT
```

Informações solicitadas:

a. Escolher outra opção.

```
{█ _ _ }
```

```
XMT{ _ }
```

```
***PREFERE ESCOLHER OUTRA OPCA0? SIM OU NAO
```

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Continuarã a execuçã da opção 22, somente se o grafo jã foi editado. Caso o grafo jã seja não orientado a rotina não serã executada e darã uma mensagem de advertência.

b. Informações finais.

Encerrada a rotina serão dadas informações sobre o número de arcos, vértices e orientação, como no exemplo adiante. É importante notar que o novo grafo será gerado sobre o grafo anterior, que será portanto destruído.

```
***0 NOVO GRAFO TEM:  
  
    16  ARCOS  
    9  VERTICES  
  
        NAO E ORIENTADO  
  
==>XMT
```

c. Matriz de adjacência.

```
{█ _ _ }          XMT{ _ }  
  
***DESEJA OBSERVAR A MATRIZ DE ADJACENCIA? SIM OU NAO  
  
    IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES
```

SIM - Mostrará no vídeo a matriz de adjacência do grafo. Se o grafo tiver mais de 15 vértices a matriz não será mostrada e será dada uma mensagem de advertência. Neste caso pode-se pedir para imprimir a matriz de adjacência, escolhendo-se depois a opção oito.

NAO - Segue a execução.

d. Número de sucessores.

{■ __}	XMT{ }
***DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES? SIM OU NAO	

SIM - Apresentará para todos os vértices o semigrau exterior de cada um.

NAO - Segue a execução.

e. Os sucessores

{■ __}	XMT{ }
***DESEJA SABER OS SUCESSORES DOS VERTICES? SIM OU NAO	

SIM - Mostrará na tela para cada vértice, além de seu semigrau exterior, os seus sucessores diretos. Se algum vértice não tem sucessores, isto será indicado. São apresentados na tela apenas três vértices de cada vez, para observar os próximos três, precionar apenas uma vez a tecla XMT no console, seguindo assim até o final.

NAO - Segue a execução.

f. Nova opção.

{█ _ _ }	XMT{ _ }
***DESEJA EXECUTAR OUTRA OPCAO? SIM OU NAO	

SIM - Mostrarã a primeira lista de opções para escolha.

NAO - Darã as mensagens de finalizaçã.

APÊNDICE B - COMO USAR OS PROGRAMAS DE APOIO

B.1 - Introdução

Os programas de apoio tem a finalidade específica de gerenciar o arquivo de grafos que pertence ao diretório SUPORTE, este arquivo é denominado ARQGRA.

São ao todo quatro programas, que devem ser utilizados pela pessoa responsável pelo arquivo, sempre que for necessária alguma alteração no mesmo.

As finalidades específicas de cada um destes programas já foram apresentadas no Capítulo II do presente trabalho.

Este apêndice tem o único objetivo de informar quais são os dados necessários para o perfeito funcionamento de cada um dos programas.

Os programas deverão estar armazenados em módulo objeto, portanto é suposto que o usuário saiba chamar um programa armazenado nesta forma.

B.2 - Programa INICIA.

Este programa não exige nenhum dado de entrada.

Sõ é usado em ocasiões muito especiais, quando se deseja destruir ou inicializar o arquivo de grafos, ARQGRA.

Este programa é de uso restrito ao responsável pelo arquivo.

B.3 - Programa LEITOR.

Permite a entrada de grafos, que serão armazenados em ARQGRA, via cartão.

Entrada dos dados:

(a) O primeiro cartão antes de cada grafo deve conter o número de vértices do grafo e se ele é orientado ou não.

Nas colunas 3 e 4: Número de vértices do grafo, que deve ser um número inteiro maior que zero e menor ou igual a 40.

Na coluna 6: Orientação do grafo.

0 - grafo não orientado

1 - grafo orientado

(b) Nos cartões seguintes são colocados o número do vértice, o número de sucessores e quais são estes sucessores, para todos os vértices do grafo.

Os números devem ser perfurados em campos de quatro colunas cada um. Estes números são inteiros, e como tal, não podem conter ponto decimal e devem estar enquadrados a direita de cada campo. Se for necessário os sucessores podem vir perfurados em um ou mais cartões de continuação, neste caso somente o primeiro cartão deste vértice conterá o número dele e o número de sucessores, os demais cartões conterão apenas os sucessores. Não é necessário que os vértices sejam colocados em ordem crescente, porém todos eles devem estar presentes, mesmo que não tenham sucessores, ou seja, que o número de sucessores seja igual a zero.

É permitido armazenar-se quantos grafos se desejar, neste caso entre um grafo e outro deve sempre existir um cartão como o indicado no ítem a.

Este programa é de uso geral, com a permissão do responsável pelo arquivo.

B.4 - Programa LISTARQ.

Este programa também não exige nenhum dado de entrada.

É usado sempre que se desejar saber o conteúdo do arquivo de grafos ARQGRA.

O programa acessa o arquivo unicamente para leitura, não fazendo portanto nenhuma alteração no mesmo.

Este programa é de uso geral.

B.5 - Programa APAGA.

Este programa permite que se retire do arquivo de grafos, ARQGRA, um ou mais grafos.

A cada grafo apagado o programa faz uma realocação dos grafos restantes, tendo-se em vista os novos espaços disponíveis.

Entrada de dados:

Deve-se entrar com o número do grafo a ser apagado nas quatro primeiras colunas do cartão, este número é inteiro, e como tal, não pode conter ponto decimal e deve vir enquadrado a direita do campo.

Existirão tantos cartões quantos forem os grafos a serem apagados.

Os grafos apagados não podem ser recuperados, a menos que sejam gravados novamente.

APÊNDICE C - CONVERSÃO PARA TERMINAIS IBM

C.1 - Introdução

Este apêndice tem o objetivo de fornecer as idéias gerais necessárias para a implantação do sistema proposto, em terminais de teleprocessamento acoplados a computadores IBM.

A conversão implicará em adaptações dos programas existentes nos diretórios SISTEMA e SUPORTE ao compilador FORTRAN disponível na instalação.

Devido a estreita semelhança entre os compiladores FORTRAN existentes nos sistemas operacionais BURROUGHS (9) e IBM (14), e principalmente pelo fato de não terem sido usadas particularidades de um ou de outro sistema, as adaptações necessárias nos programas poderão ser feitas com relativa facilidade.

C.2 - Opções de OS/VS2 (15)

Serão analisadas duas opções de instalação do sistema operacional da IBM, OS/VS2.

(a) MVS (Multiple Virtual System)

Esta opção possibilita uma boa flexibilidade de programação, usando-se comandos de terminal. Estes comandos chamados CLIST (Command List) permitem, entre outras funções: a leitura e impressão na tela, executar funções lógicas e chamar programas armazenados. Usando-se assim estes comandos pode-se elaborar programas que permitem ao usuário comunicar-se com o sistema, estes programas são ditos Command Procedures.

(b) SVS (Simple Virtual System)

Esta opção do sistema operacional não apresenta as facilidades disponíveis em MVS para as Command Procedures.

C.3 - Implantação do SISGRAFO

(a) Sistema MVS

Pode-se construir uma Command Procedure que conteria: as mensagens de inicialização e finalização, as informações sobre as opções e as listas de opções.

A leitura da opção escolhida seria feita dentro desta Command Procedure que chamaria por um comando CALL um programa que executaria a opção desejada. Como neste caso as opções serão programas completamente isolados, porém usarão sempre o grafo montado por uma das opções de edição, estas opções (de edição) deverão gravar o grafo editado em um arquivo temporário, que seria apagado ao final da execução, ou alterado caso seja chamada uma nova opção de edição.

No Capítulo II, páginas 12 e 13, deste trabalho foi feita uma divisão do programa SISGRAFO em três partes fundamentais, a Command Procedure referida acima faria as funções indicadas nas partes 1 e 2; a parte 3 seria subdividida em programas independentes.

(b) Sistema SVS

Neste caso, após as devidas adaptações ao Compilador disponível, os módulos de programação existentes no SISGRAFO poderão ser implantados sem grandes alterações estruturais, uma vez que não será possível a utilização de uma Command Procedure. Isto implica em que, todo o controle sobre a formatação da tela deverá ser feito pelos módulos de programação.

C.4 - Programas de Apoio

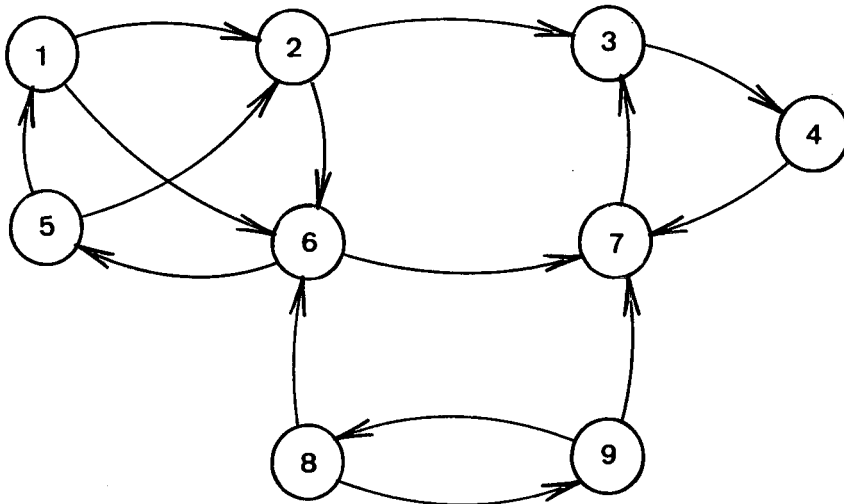
Para estes programas do diretório suporte somente seriam necessárias as adaptações ao compilador FORTRAN disponível.

APÊNDICE D - TESTES EXEMPLO

Neste apêndice estão ilustradas quatro execuções do Sistema. São apresentadas as listagens geradas, uma vez que foi solicitada a impressão de um relatório final.

Os grafos armazenados no arquivo ARQGRA usados neste apêndice são os seguintes:

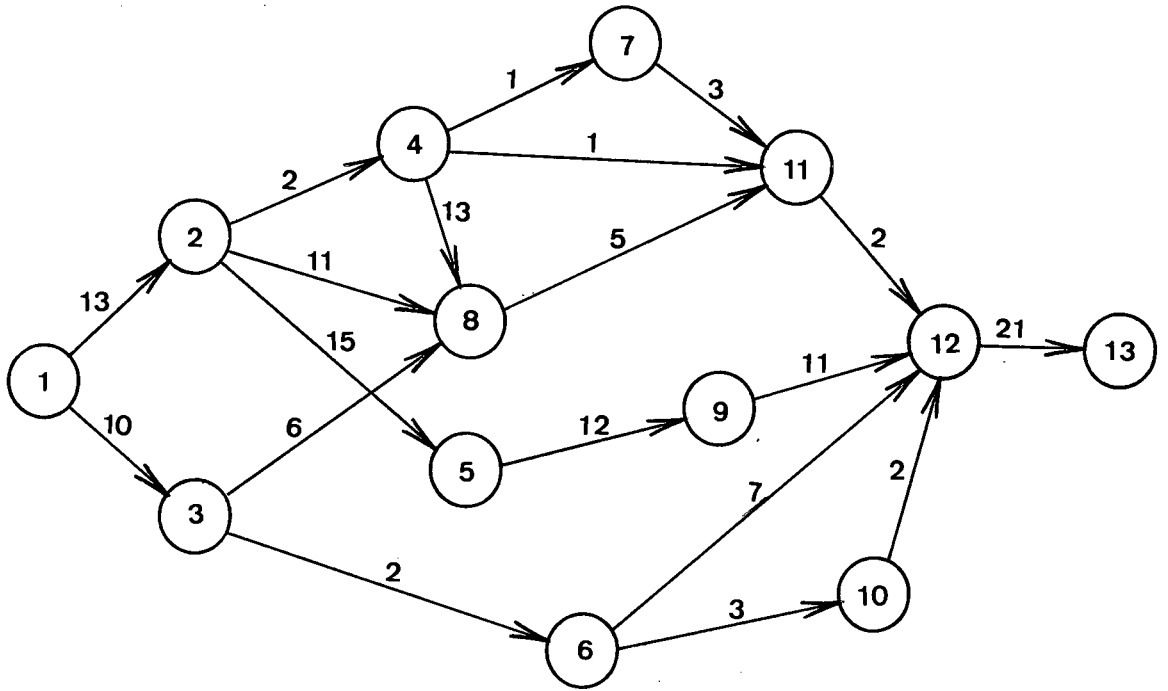
(a) Grafo número 3:



(b) Grafo número 6:

Este grafo é o mesmo utilizado como exemplo no Apêndice A, página 149.

(c) Grafo número 7:



```
*****  
*                               *  
*           SISGRAFO           *  
*                               *  
*           SISTEMA DIDATICO   *  
*                               *  
*           PARA SOLUCAO DE PROBLEMAS *  
*                               *  
*           ENVOLVENDO A TEORIA DOS GRAFOS *  
*                               *  
*****
```

*** A SIGLA XMT INDICA UMA PARADA PARA OBSERVACAO DA TELA
PARA SEGUIR PRESSIONE A TECLA XMT

*** ENTRE COM SEU NOME E NUMERO

RELATORIO DO ALUNO

FLORIANO BRANDAO FILHO - TESTE

NUMERO 1

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 1

GERA UM GRAFO ALEATORIAMENTE

PODE GERAR GRAFOS ORIENTADOS OU NAO

O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)

A SEMENTE PARA GERACAO PODE SER QUALQUER NUMERO

A PROBABILIDADE DEVE SER > ZERO E <=100

PODE-SE LIMITAR O NUMERO DE ARCOS DESEJADOS
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR
APOS A GERACAO JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA
O SEMIGRAU EXTERIOR DE CADA VERTICE
OS SUCESSORES DE CADA VERTICE

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** O GRAFO E ORIENTADO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ENTRE COM O NUMERO DE VERTICES DO GRAFO, MAXIMO 40

*** VALOR LIDO = 10

*** ENTRE COM O NUMERO DE ARCOS/ARESTAS DO GRAFO
CASO NAO DESEJE LIMITAR ENTRE COM ZERO

*** VALOR LIDO = 94

*** NUMERO MAXIMO DE ARCOS/ARESTAS PARA SEU GRAFO E 90
ENTRE NOVAMENTE

*** ENTRE COM O NUMERO DE ARCOS/ARESTAS DO GRAFO
CASO NAO DESEJE LIMITAR ENTRE COM ZERO

*** VALOR LIDO = 82

*** ENTRE COM A SEMENTE ALEATORIA

*** VALOR LIDO = 12.34

*** O GRAFO GERADO TEM:
82 ARCOS/ARESTAS
10 VERTICES
E ORIENTADO

*** PREFERE GERAR UM GRAFO DIFERENTE ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ? SIM OU NAO
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES

*** INFORMACAO LIDA = SIM

*** MATRIZ DE ADJACENCIA DO GRAFO 10 X 10

	1	2	3	4	5	6	7	8	9	10
1	0	1	1	1	0	1	1	0	1	1
2	1	0	1	1	1	1	1	1	1	1
3	1	0	0	1	1	1	1	1	1	0
4	1	1	1	0	1	1	1	1	1	1
5	1	0	1	1	0	1	1	1	1	1
6	1	1	1	1	0	0	1	1	1	1
7	1	1	1	1	1	1	0	1	1	1
8	1	1	1	1	1	1	1	0	1	0
9	1	1	1	1	1	1	1	1	0	1
10	1	1	1	1	1	1	1	1	0	0

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** SUCESSORES DOS VERTICES

*** VERTICE 1 SEMIGRAU EXTERIOR 7

SUCCESSORES 2 3 4 6 7 9 10

*** VERTICE 2 SEMIGRAU EXTERIOR 9

SUCCESSORES 1 3 4 5 6 7 8 9 10

*** VERTICE 3 SEMIGRAU EXTERIOR 7

SUCCESSORES 1 4 5 6 7 8 9

*** SUCESSORES DOS VERTICES

*** VERTICE 4 SEMIGRAU EXTERIOR 9

SUCCESSORES 1 2 3 5 6 7 8 9 10

*** VERTICE 5 SEMIGRAU EXTERIOR 8

SUCCESSORES 1 3 4 6 7 8 9 10

*** VERTICE 6 SEMIGRAU EXTERIOR 8
SUCESSORES 1 2 3 4 7 8 9 10

*** SUCESSORES DOS VERTICES

*** VERTICE 7 SEMIGRAU EXTERIOR 9
SUCESSORES 1 2 3 4 5 6 8 9 10

*** VERTICE 8 SEMIGRAU EXTERIOR 8
SUCESSORES 1 2 3 4 5 6 7 9

*** VERTICE 9 SEMIGRAU EXTERIOR 9
SUCESSORES 1 2 3 4 5 6 7 8 10

*** SUCESSORES DOS VERTICES

*** VERTICE 10 SEMIGRAU EXTERIOR 8
SUCESSORES 1 2 3 4 5 6 7 8

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 10

MOSTRA NA TELA A PROXIMA LISTA DE OPCOES
A PROXIMA LISTA SERA A SEGUNDA
PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA
MESMO QUE NAO CONSTE DA LISTA APRESENTADA

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 02):

- 11 - VALORAR UM GRAFO
- 12 - CAMINHO MINIMO (FLOYD)
- 13 - DETERMINA A EXISTENCIA DE CIRCUITO (DEMOUCRON)
- 14 - CAMINHO MAXIMO (FLOYD)
- 15 - RAO, DIAMETRO (GRAFOS NAO VALORADOS)
- 16 - COMPONENTES F-CONEXAS (MALGRANGE)
- 17 - RAO, DIAMETRO (GRAFOS VALORADOS)
- 18 - FLUXO MAXIMO (FORD - FULKERSON)
- 19 - CAMINHO MINIMO (DIJKSTRA)
- 20 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 15

RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAMENTOS
VALIDA PARA GRAFOS NAO VALORADOS
CALCULA O RAO E O DIAMETRO INTERIORES E EXTERIORES
DETERMINA O CENTRO E VERTICES PERIFERICOS
DETERMINA OS AFASTAMENTOS DOS VERTICES
CALCULA A MATRIZ DE DISTANCIAS
PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS
VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E AFASTAMENTOS ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** INFORMACOES SOBRE O GRAFO

RAIO EXTERIOR 1

DIAMETRO EXTERIOR 2

CENTROS EXTERIORES 2 4 7 9

PERIFERICOS EXTERIORES 1 3 5 6 8 10

RAIO INTERIOR 1

DIAMETRO INTERIOR 2

CENTROS INTERIORES 1 3 4 6 7

PERIFERICOS INTERIORES 2 5 8 9 10

CENTROS BI-DIRECIONAIS 4 7

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** VOCE EXECUTOU 1 REENTRADAS

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA

APOS ENTRE BYE

*** INICIO DIA 05/01/80 HORA 11:23

*** TERMINO DIA 05/01/80 HORA 11:31

```
*****  
*                               *  
*           SISGRAFO           *  
*                               *  
*       SISTEMA DIDATICO       *  
*                               *  
*   PARA SOLUCAO DE PROBLEMAS  *  
*                               *  
*   ENVOLVENDO A TEORIA DOS GRAFOS *  
*                               *  
*****
```

*** A SIGLA XMT INDICA UMA PARADA PARA OBSERVACAO DA TELA
PARA SEGUIR PRESSIONE A TECLA XMT

*** ENTRE COM SEU NOME E NUMERO

RELATORIO DO ALUNO

FLORIANO BRANDAO FILHO - TESTE

NUMERO 2

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 2

CHAMA UM GRAFO ARMAZENADO ANTERIORMENTE

E NECESSARIO SABER O NUMERO DO GRAFO A SER USADO
TODO GRAFO ARMAZENADO E ORIENTADO
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR

APOS A LEITURA JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE

OS SUCESSORES DE CADA VERTICE

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ENTRE COM O NUMERO DO GRAFO PADRONIZADO

*** NUMERO DO GRAFO A SER USADO = 6

*** O GRAFO CHAMADO TEM:

17 ARCOS/ARESTAS

9 VERTICES

E ORIENTADO

*** PREFERE CHAMAR OUTRO GRAFO PADRONIZADO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ? SIM OU NAO

IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES

*** INFORMACAO LIDA = SIM

*** MATRIZ DE ADJACENCIA DO GRAFO 9 X 9

	1	2	3	4	5	6	7	8	9
1	0	1	1	0	1	0	0	0	0
2	0	0	1	0	1	1	0	0	0
3	0	0	0	1	0	0	0	0	0
4	0	0	0	0	0	0	1	1	0
5	0	0	1	0	0	0	1	0	0
6	0	0	0	0	1	0	1	1	0
7	0	0	0	0	0	0	0	0	1
8	0	0	0	0	0	1	0	0	1
9	0	0	0	0	0	0	0	0	0

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01)

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 11

PERMITE VALORAR UM GRAFO

PODE-SE DAR UM VALOR CONSTANTE PARA TODOS ARCOS/ARESTAS

GRAFOS NAO ORIENTADOS TERAO OS VALORES SIMETRIZADOS

PODE-SE PEDIR UMA SAIDA DA MATRIZ DE VALORES

A IMPRESSAO CONTEM DUAS CASAS DECIMAIS

VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

O MAXIMO VALOR DEVERA SER MENOR QUE 10 ELEVADO A 8

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA DAR UM VALOR CONSTANTE A TODOS ARCOS/ARESTAS ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(1- 2)(1- 3)(1- 5)(2- 3)(2- 5)(2- 6)(3- 4)(4- 7)(4- 8)

*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(5- 3)(5- 7)(6- 5)(6- 7)(6- 8)(7- 9)(8- 6)(8- 9)

*** SEU GRAFO JA FOI VALORADO

*** PREFERE UMA VALORACAO DIFERENTE ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE VALORES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 17

RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAMENTOS

VALIDA PARA GRAFOS VALORADOS

CALCULA O RAO E O DIAMETRO INTERIORES E EXTERIORES

DETERMINA OS CENTROS E VERTICES PERIFERICOS

DETERMINA OS AFASTAMENTOS DOS VERTICES

CALCULA A MATRIZ DE DISTANCIAS

PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS

VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA DAR VALORES AOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

VALORES LIDOS PARA OS VERTICES

VERT	VALOR	VERT	VALOR	VERT	VALOR	VERT	VALOR
1	345.00	2	300.00	3	45.00	4	456.00
5	0.03	6	67.00	7	34.00	8	35.00
9	367.00						

*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E AFASTAMENTOS ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** INFORMACOES SOBRE O GRAFO

RAIO EXTERIOR 7707.00

DIAMETRO EXTERIOR *****

CENTROS EXTERIORES 1

PERIFERICOS EXTERIORES 2 3 4 5 6 7 8 9

RAIO INTERIOR 7245.00

DIAMETRO INTERIOR *****

CENTROS INTERIORES 9

PERIFERICOS INTERIORES 1 2 3 4 5 6 7 8

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 18

ALGORITMO DE FORD-FULKERSON (FLUXO MAXIMO)

VALIDA PARA GRAFOS VALORADOS

INDICA O VALOR DO FLUXO MAXIMO

IMPRIME OS ARCOS PARA O FLUXO MAXIMO

SAO INDICADOS OS ARCOS SATURADOS

O FLUXO E DETERMINADO ENTRE O VERTICE 1 E NVER

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** O FLUXO MAXIMO OBTIDO POR FORD-FULKERSON FOI
20.00

*** DESEJA IMPRIMIR OS VALORES DOS ARCOS
PARA O FLUXO MAXIMO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

VALORES DOS ARCOS PARA O FLUXO MAXIMO (FORD-FULKERSON)

ARCO	1	2	=	6.00	
ARCO	1	3	=	4.00	SATURADO
ARCO	1	5	=	10.00	SATURADO
ARCO	2	3	=	2.00	SATURADO
ARCO	2	5	=	3.00	SATURADO
ARCO	2	6	=	1.00	
ARCO	3	4	=	6.00	
ARCO	4	7	=	2.00	SATURADO
ARCO	4	8	=	4.00	
ARCO	5	7	=	13.00	
ARCO	6	7	=	1.00	SATURADO
ARCO	7	9	=	16.00	SATURADO
ARCO	8	9	=	4.00	SATURADO

*** FORAM IMPRESSOS OS RESULTADOS PEDIDOS

O ESCANINHO TEM O NUMERO DO TERMINAL EM USO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 12

ALGORITMO DE FLOYD PARA CAMINHO MINIMO

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

CALCULA A MATRIZ DE DISTANCIA MINIMA DO GRAFO

CALCULA A MATRIZ DE ROTEAMENTO DO GRAFO

PODE-SE IMPRIMIR A MATRIZ DE DISTANCIA E DE ROTEAMENTO

PODE-SE PERGUNTAR A DISTANCIA MINIMA ENTRE DOIS VERTICES

OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIA ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE ROTEAMENTO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

MATRIZ DE ROTEAMENTO MINIMO DO GRAFO 9 X 9

	1	2	3	4	5	6	7	8	9
1	1	2	3	3	5	2	3	3	3
2	0	2	3	3	5	6	6	6	6
3	0	0	3	4	4	4	4	4	4
4	0	0	8	4	8	8	7	8	8
5	0	0	3	3	5	3	3	3	3
6	0	0	5	5	5	6	7	8	8
7	0	0	0	0	0	0	7	0	9
8	0	0	6	6	6	6	6	8	9
9	0	0	0	0	0	0	0	0	9

*** FORAM IMPRESSOS OS RESULTADOS PEDIDOS

O ESCANINHO TEM O NUMERO DO TERMINAL EM USO

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VERTICES DO CAMINHO MINIMO ENTRE (1 E 9)

1 3 4 8 9

*** DISTANCIA MINIMA ENTRE OS VERTICES (1 E 9) 21.00

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VERTICES DO CAMINHO MINIMO ENTRE (1 E 8)

1 3 4 8

*** DISTANCIA MINIMA ENTRE OS VERTICES (1 E 8) 17.00

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VERTICES DO CAMINHO MINIMO ENTRE (1 E 7)

1 3 4 7

*** DISTANCIA MINIMA ENTRE OS VERTICES (1 E 7) 13.00

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 19

ALGORITMO DE DIJKSTRA(CAMINHO MINIMO)

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

DEVE-SE INDICAR O VERTICE RAIZ

FORNECE A DISTANCIA DA RAIZ A TODOS OS VERTICES

PODE-SE ESCOLHER QUALQUER VERTICE PARA RAIZ

USAR FLOYD PARA SABER OS VERTICES DO CAMINHO

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ENTRE COM O VALOR DA RAIZ DESEJADA

*** DISTANCIAS ENTRE A RAIZ 1 E OS VERTICES:

VERTICE	1 ...	0.00
VERTICE	2 ...	9.00
VERTICE	3 ...	4.00
VERTICE	4 ...	11.00
VERTICE	5 ...	10.00
VERTICE	6 ...	15.00
VERTICE	7 ...	13.00
VERTICE	8 ...	17.00
VERTICE	9 ...	21.00

*** PARA SABER OS VERTICES DO CAMINHO USAR FLOYD (OPCAO=12)

*** DESEJA MUDAR O VALOR DA RAIZ ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ENTRE COM O VALOR DA RAIZ DESEJADA

*** DISTANCIAS ENTRE A RAIZ 2 E OS VERTICES:

VERTICE 1 ...	*****
VERTICE 2 ...	0.00
VERTICE 3 ...	2.00
VERTICE 4 ...	9.00
VERTICE 5 ...	3.00
VERTICE 6 ...	6.00
VERTICE 7 ...	7.00
VERTICE 8 ...	11.00
VERTICE 9 ...	15.00

*** PARA SABER OS VERTICES DO CAMINHO USAR FLOYD (OPCAO=12)

*** DESEJA MUDAR O VALOR DA RAIZ ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** PARABENS, VOCE NAO NECESSITOU NENHUMA REENTRADA

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA

APOS ENTRE BYE

*** INICIO DIA 10/10/79 HORA 20:23

*** TERMINO DIA 10/10/79 HORA 20:35

```
*****  
*                               *  
*           SISGRAFO           *  
*                               *  
*           SISTEMA DIDATICO   *  
*                               *  
*           PARA SOLUCAO DE PROBLEMAS *  
*                               *  
*           ENVOLVENDO A TEORIA DOS GRAFOS *  
*                               *  
*****
```

*** A SIGLA XMT INDICA UMA PARADA PARA OBSERVACAO DA TELA
PARA SEGUIR PRESSIONE A TECLA XMT

*** ENTRE COM SEU NOME E NUMERO

RELATORIO DO ALUNO

FLORIANO BRANDAO FILHO - TESTE

NUMERO 3

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 2

CHAMA UM GRAFO ARMAZENADO ANTERIORMENTE

E NECESSARIO SABER O NUMERO DO GRAFO A SER USADO
TODO GRAFO ARMAZENADO E ORIENTADO
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 14

ALGORITMO DE FLOYD PARA CAMINHO MAXIMO

SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS

O GRAFO NAO PODE TER NENHUM CIRCUITO

CALCULA O CAMINHO CRITICO NUMA REDE PERT

CALCULA A MATRIZ DE DISTANCIA MAXIMA DO GRAFO

CALCULA A MATRIZ DE ROTEAMENTO MAXIMO DO GRAFO

PODE-SE IMPRIMIR QUALQUER DAS MATRIZES

PODE-SE PEDIR A DISTANCIA MAXIMA ENTRE DOIS VERTICES

OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** O SEU GRAFO AINDA NAO FOI VALORADO

A ROTINA DE VALORIZACAO SERA CHAMADA

POREM SERA INCREMENTADO UM ERRO AO CONTADOR

*** DESEJA DAR UM VALOR CONSTANTE A TODOS ARCOS/ARESTAS ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(1- 2)(1- 3)(2- 4)(2- 5)(2- 8)(3- 6)(3- 8)(4- 7)(4- 8)

*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(4- 11)(5- 9)(6- 10)(6- 12)(7- 11)(8- 11)(9- 12)(10- 12)(11- 12)

*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS ABAIXO:

(12- 13)

*** SEU GRAFO JA FOI VALORADO

*** PREFERE UMA VALORACAO DIFERENTE ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE VALORES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS NIVEIS EXISTENTES NO GRAFO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** NIVEIS EXISTENTES NO GRAFO

NIVEL 1 ...	1			
NIVEL 2 ...	2	3		
NIVEL 3 ...	4	5	6	
NIVEL 4 ...	7	8	9	10
NIVEL 5 ...	11			
NIVEL 6 ...	12			
NIVEL 7 ...	13			

*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIA ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE ROTEAMENTO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

MATRIZ DE ROTEAMENTO MAXIMO DO GRAFO 13 X 13

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	2	3	2	2	3	2	2	2	3	2	2	2
2	0	2	0	4	5	0	4	4	5	0	4	5	5
3	0	0	3	0	0	6	0	8	0	6	8	8	8
4	0	0	0	4	0	0	7	8	0	0	8	8	8
5	0	0	0	0	5	0	0	0	9	0	0	9	9
6	0	0	0	0	0	6	0	0	0	10	0	12	12
7	0	0	0	0	0	0	7	0	0	0	11	11	11
8	0	0	0	0	0	0	0	8	0	0	11	11	11
9	0	0	0	0	0	0	0	0	9	0	0	12	12
10	0	0	0	0	0	0	0	0	0	10	0	12	12
11	0	0	0	0	0	0	0	0	0	0	11	12	12
12	0	0	0	0	0	0	0	0	0	0	0	12	13
13	0	0	0	0	0	0	0	0	0	0	0	0	13

*** FORAM IMPRESSOS OS RESULTADOS PEDIDOS

O ESCANINHO TEM O NUMERO DO TERMINAL EM USO

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VERTICES DO CAMINHO MAXIMO ENTRE (1 E 13)

1 2 5 9 12 13

*** DISTANCIA MAXIMA ENTRE OS VERTICES (1 E 13) 72.00

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** INDIQUE ENTRE QUE CERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VALORES LIDOS I=15 J=13 INCOMPATIVEL COM NUMERO DE VERTICES
TENTE NOVAMENTE

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** VERTICES DO CAMINHO MAXIMO ENTRE (2 E 11)

2 4 8 11

*** DISTANCIA MAXIMA ENTRE OS VERTICES (2 E 11) 20.00

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** INDIQUE ENTRE QUE VERTICES I-J DESEJA O CAMINHO MINIMO (MAXIMO)

*** NAO EXISTE NENHUM CAMINHO ENTRE OS VERTICES (3 E 9)

*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 22

DESORIENTAR UM GRAFO

ALTERA O GRAFO ORIGINAL

O GRAFO PASSA A SER NAO ORIENTADO

SE HOVER O ARCO (I,J) HAVERA O ARCO (J,I)

NOS GRAFOS VALORADOS $VAL(I,J) = VAL(J,I)$

PARA VALORES NOS DOIS SENTIDOS TEM-SE:

$$VAL(I,J) = MAX(VAL(I,J),VAL(J,I))$$

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** O NOVO GRAFO TEM:

19 ARCOS

13 VERTICES

NAO E ORIENTADO

*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ? SIM OU NAO

IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES

*** INFORMACAO LIDA = NAO

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

0 - INTERROMPER OS TRABALHOS

1 - GERAR UM GRAFO ALEATORIO

2 - CHAMAR GRAFO ARMAZENADO

3 - FORNECER GRAFO PELO TERMINAL

4 - ARMAZENAR GRAFO

5 - GRAFO PARTE FORNECIDO PARTE GERADO

6 - SUCESSORES DOS VERTICES

7 - SEMIGRAU EXTERIOR DE CADA VERTICE

8 - IMPRIMIR MATRIZ DE ADJACENCIA

9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES

10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 21

ALGORITMO DE KRUSKAL (ARVORE PARCIAL MINIMA)

VALIDA SO PARA GRAFOS NAO ORIENTADOS E VALORADOS

FORNECE OS ARCOS DA ARVORE PARCIAL

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS ARCOS DA ARVORE PARCIAL MINIMA ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ARCOS DA ARVORE PARCIAL MINIMA (KRUSKAL)

ARCO	4 - 7	1.00
ARCO	4 -11	1.00
ARCO	2 - 4	2.00
ARCO	3 - 6	2.00
ARCO	10 -12	2.00
ARCO	11 -12	2.00
ARCO	6 -10	3.00
ARCO	8 -11	5.00
ARCO	1 - 3	10.00
ARCO	9 -12	11.00
ARCO	5 - 9	12.00
ARCO	12 -13	21.00

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 2

CHAMAR UM GRAFO ARMAZENADO ANTERIORMENTE
E NECESSARIO SABER O NUMERO DO GRAFO A SER USADO
TODO GRAFO ARMAZENADO E ORIENTADO
ESTA E UMA ROTINA DE EDICAO DE UM GRAFO
PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR

APOS A LEITURA JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE

OS SUCESSORES DE CADA VERTICE

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ENTRE COM O NUMERO DO GRAFO PADRONIZADO

*** ENTRADA ERRADA, TENDE NOVAMENTE

*** ENTRE COM O NUMERO DO GRAFO PADRONIZADO

*** NUMERO DO GRAFO A SER USADO = 3

*** O GRAFO CHAMADO TEM:

15 ARCOS/ARESTAS

9 VERTICES

E ORIENTADO

*** PREFERE CHAMAR OUTRO GRAFO PADRONIZADO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ? SIM OU NAO

IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES

*** INFORMACAO LIDA = SIM

*** MATRIZ DE ADJACENCIA DO GRAFO 9 X 9

	1	2	3	4	5	6	7	8	9
1	0	1	0	0	0	1	0	0	0
2	0	0	1	0	0	1	0	0	0
3	0	0	0	1	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0
5	1	1	0	0	0	0	0	0	0
6	0	0	0	0	1	0	1	0	0
7	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	1	0	0	1
9	0	0	0	0	0	0	1	1	0

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 16

DECOMPOSICAO EM COMPONENTES F-CONEXAS (MALGRANGE)

DETERMINA AS COMPONENTES F-CONEXAS

INDICA SE O GRAFO E F-CONEXO

UTILIZA O ALGORITMO DE MALGRANGE

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** COMPONENTES F-CONEXAS DO GRAFO

VERTICES ...	8	9		
VERTICES ...	3	4	7	
VERTICES ...	1	2	5	6

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** VOCE EXECUTOU 3 REENTRADAS

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA
APOS ENTRE BYE

*** INICIO DIA 26/01/80 HORA 11:44

*** TERMINO DIA 26/01/80 HORA 12:17


```
*****
*                               *
*           SISGRAFO           *
*                               *
*           SISTEMA DIDATICO   *
*                               *
*           PARA SOLUCAO DE    *
*           PROBLEMAS          *
*                               *
*           ENVOLVENDO A      *
*           TEORIA DOS GRAFOS *
*                               *
*****
```

*** A SIGLA XMT INDICA UMA PARADA PARA OBSERVACAO DA TELA
PARA SEGUIR PRESSIONE A TECLA XMT

*** ENTRE COM SEU NOME E NUMERO

RELATORIO DO ALUNO

FLORIANO BRANDAO FILHO - TESTE

NUMERO 4

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 1

GERA UM GRAFO ALEATORIAMENTE

PODE GERAR GRAFOS ORIENTADOS OU NAO

O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)

A SEMENTE PARA GERACAO PODE SER QUALQUER NUMERO

A PROBABILIDADE DEVE SER > ZERO E <=100

PODE-SE LIMITAR O NUMERO DE ARCOS DESEJADOS

ESTA E UMA ROTINA DE EDICAO DE UM GRAFO

PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR
APOS A GERACAO JA ESTAO A DISPOSICAO:

A MATRIZ DE ADJACENCIA

O SEMIGRAU EXTERIOR DE CADA VERTICE

OS SUCESSORES DE CADA VERTICE

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** O GRAFO E ORIENTADO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ENTRE COM O NUMERO DE VERTICES DO GRAFO, MAXIMO 40

*** VALOR LIDO = 20

*** ENTRE COM O NUMERO DE ARCOS/ARESTAS DO GRAFO
CASO NAO DESEJE LIMITAR ENTRE COM ZERO

*** VALOR LIDO = 0

*** ENTRE COM A SEMENTE ALEATORIA

*** VALOR LIDO = 7569.70

*** ENTRE COM A PROBABILIDADE DA EXISTENCIA DE ARCOS/ARESTAS

*** PROBABILIDADE LIDA = 0.50

*** O GRAFO GERADO TEM:

188 ARCOS/ARESTAS

20 VERTICES

E ORIENTADO

*** PREFERE GERAR UM GRAFO DIFERENTE ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ? SIM OU NAO
IMPORTANTE: SO VALIDO PARA GRAFOS ATE 15 VERTICES

*** INFORMACAO LIDA = SIM

*** NUMERO DE VERTICES = 20 MATRIZ NAO PODE SER OBSERVADA NA TELA

*** DESEJA SABER O SEMIGRAU EXTERIOR DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** TOTAL DE SUCESSORES DOS VERTICES

VERT	NSUC	VERT	NSUC	VERT	NSUC	VERT	NSUC
1	9	6	10	11	9	16	8
2	9	7	6	12	10	17	12
3	9	8	11	13	11	18	9
4	9	9	13	14	12	19	9
5	9	10	7	15	6	20	10

*** DESEJA SABER OS SUCESSORES DOS VERTICES ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = SIM

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01):

- 0 - INTERROMPER OS TRABALHOS
- 1 - GERAR UM GRAFO ALEATORIO
- 2 - CHAMAR GRAFO ARMAZENADO
- 3 - FORNECER GRAFO PELO TERMINAL
- 4 - ARMAZENAR GRAFO
- 5 - GRAFO PARTE FORNECIDO PARTE GERADO
- 6 - SUCESSORES DOS VERTICES
- 7 - SEMIGRAU EXTERIOR DE CADA VERTICE
- 8 - IMPRIMIR MATRIZ DE ADJACENCIA
- 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICES
- 10 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 10

MOSTRA NA TELA A PROXIMA LISTA DE OPCOES

A PROXIMA LISTA SERA A SEGUNDA

PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA

MESMO QUE NAO CONSTE DA LISTA APRESENTADA

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 02):

- 11 - VALORAR UM GRAFO
- 12 - CAMINHO MINIMO (FLOYD)
- 13 - DETERMINA A EXISTENCIA DE CIRCUITO (DEMOUCRON)
- 14 - CAMINHO MAXIMO (FLOYD)
- 15 - RAI0, DIAMETRO (GRAFOS NAO VALORADOS)
- 16 - COMPONENTES F-CONEXAS (MALGRANGE)
- 17 - RAI0, DIAMETRO (GRAFOS VALORADOS)
- 18 - FLUXO MAXIMO (FORD-FULKERSON)
- 19 - CAMINHO MINIMO (DIJKSTRA)
- 20 - PROXIMA LISTA DE OPCOES

*** OPCAO LIDA = 15

RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAMENTOS

VALIDA PARA GRAFOS NAO VALORADOS

CALCULA O RAI0 E O DIAMETRO INTERIORES E EXTERIORES

DETERMINA O CENTRO E VERTICES PERIFERICOS

DETERMINA OS AFASTAMENTOS DOS VERTICES

CALCULA A MATRIZ DE DISTANCIAS

PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS

VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS (*)

*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E AFASTAMENTOS ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** INFORMACOES SOBRE O GRAFO

RAIO EXTERIOR 2

DIAMETRO EXTERIOR 3

CENTROS EXTERIORES 1 2 3 4 5 6 7 8 9 10
11 12 13 14 17 18 19 20

PERIFERICOS EXTERIORES 15 16

RAIO INTERIOR 2

DIAMETRO INTERIOR 3

CENTROS INTERIORES 1 3 4 5 6 7 8 10 11 12
13 14 15 16 17 18 19 20

PERIFERICOS INTERIORES	2	9								
CENTROS BI-DIRECIONAIS	1	3	4	5	6	7	8	10	11	12
	13	14	17	18	19	20				

*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO

*** INFORMACAO LIDA = NAO

*** VOCE EXECUTOU 1 REENTRADAS

*** FIM ***

*** AGUARDE MENSAGEM DO SISTEMA
APOS ENTRE BYE

*** INICIO DIA 26/01/80 HORA 14: 8

*** TERMINO DIA 26/01/80 HORA 14:14

APÊNDICE E - LISTAGENS DOS MÓDULOS DE PROGRAMAÇÃO

Neste apêndice estão listados os programas contidos nos dois di-
retórios:

SISTEMA/SISGRAFO

SUPORTE	}	INICIA
		LEITOR
		LISTARQ
		APAGA

SISTEMA/SISGRAFO

```
$RESET FREE
FILE 8=RELAT,UNIT=PRINTER
FILE 15(KIND=DISK,TITLE="SISTEMA/ARQGRA",MAXRECSIZE=1,BLOCKSIZE
=30)
FILE 5(KIND=DC,MAXRECSIZE=14)
INTEGER TOP
DIMENSION MADJ(40,40),NSUC(40)
DIMENSION NOME(40)
DIMENSION VAL(40,40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK3/TOP,MAX
COMMON /BLK4/IVLR,VAL
C
C ***** PROGRAMA PRINCIPAL *****
C
CALL DIAHOR(DI1,DI2,IH1,IMI)
25 FORMAT(1H1,////////////////////////////////////)
MAX=10000
IXIST=0
NERRO=0
WRITE(6,10)
10 FORMAT(1H1,///20X,40('*'),/20X,1H*,38(' '),1H*/20X,1H*,15
(' '),1S
*ISGRAFO',15(' '),1H*/20X,1H*,38(' '),1H*/20X,1H*,10(' ')
,'SISTEM
*A DIDATICO',10(' '),1H*/20X,1H*,38(' '),1H*/20X,1H*,5X
,'PARA
*SOLUCAO DE PROBLEMAS',5X,1H*/20X,1H*,38(' '),1H*/20X,1
H*,2X,'E
*INVOLVENDO A TEORIA DOS GRAFOS ',1H*/20X,1H*,38(' '),
1H*/20X
*,40('*'),//5X,'*** A SIGLA XMT INDICA UMA PARADA PARA OBSE
RVACAO D
*A TELA',/9X,'PARA SEGUIR PRESSIONE A TECLA XMT')
CALL PAUSA
16 WRITE(6,25)
WRITE(6,15)
15 FORMAT(/5X,'*** ENTRE COM SEU NOME E NUMERO')
CALL LINHA(1)
READ(5,20,DATA=99)NOME,NUMERO
20 FORMAT(40A1,I3)
GO TO 603
99 CALL MSG(620,IREL)
NERRO=NERRO+1
```

```
CALL PAUSA
GO TO 16
603 WRITE(6,21)
21  FORMAT(1H1,////////////////////////////////////
*5X,'*** DESEJA RELATORIO FINAL IMPRESSO ? SIM OU NAO')
CALL SINO(IREI,8601,8602,8603)
601 WRITE(8,10)
WRITE(8,15)
WRITE(8,22)NOME,NUMERO
22  FORMAT(///20X,'RELATORIO DO ALUNO',///10X,40A1,5X,'NUME
RO',15)
602 CALL OPCAO
IF(NERRO.EQ.0)GO TO 501
WRITE(6,6)NERRO
IF(IREL.EQ.1)WRITE(8,6)NERRO
6  FORMAT(1H1,///5X,'*** VOCE EXECUTOU',16,2X,'REENTRADAS')
GO TO 502
501 CALL MSG(600,IREL)
502 CALL MSG(601,IREL)
CALL DIAHOR(DT1,DT2,IHT,IMT)
WRITE(6,17)DI1,DI2,IHI,IMI,DT1,DT2,IHT,IMT
IF(IREL.EQ.1)WRITE(8,17)DI1,DI2,IHI,IMI,DT1,DT2,IHT,IMT
17  FORMAT(///5X,'*** INICIO ..... DIA',2X,2A6,' HORA ',I2,
':',I2, /
* /5X,'*** TERMINO .... DIA',2X,2A6,' HORA ',I2,':',I2)
CALL EXIT
END
```

```
C
C  PROGRAMA PARA OBTER A DATA E HORA
C
```

```
SUBROUTINE DIAHOR(D1,D2,IHOR,IMIN)
DIMENSION DIA(2)
DIA(1)=' / /'
DIA(2)=' '
B=TIME(15)
DIA(1)=CONCAT(DIA(1),B,23,47,16)
DIA(1)=CONCAT(DIA(1),B,47,31,16)
DIA(2)=CONCAT(DIA(2),B,47,15,16)
D1=DIA(1)
D2=DIA(2)
MHO=TIME(1)
TEMP=FLOAT(MHO)/216000.
IHOR=IFIX(TEMP)
IMIN=IFIX((TEMP-FLOAT(IHOR))*60)
RETURN
```


END

C
C
C

DETERMINA QUAL OPCAO A ESCOLHER

SUBROUTINE OPCAO

INTEGER TOP

DIMENSION MADJ(40,40),NSUC(40)

DIMENSION VAL(40,40)

COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC

COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC

COMMON /BLK3/TOP,MAX

COMMON /BLK4/IVLR,VAL

DATA SENHA/'SHWA'/

25 FORMAT(1H1,////////////////////////////////////)

3000 CALL MSG(591,IREL)

GO TO 650

1000 CALL MSG(592,IREL)

GO TO 650

2000 CALL MSG(593,IREL)

650 CALL LINHA(2)

READ(5,10,DATA=99)IOPC

10 FORMAT(I2)

IF(IOPC.NE.0)GO TO 750

CALL MSG(602,IREL)

CALL PAUSA

RETURN

750 IF(IOPC.LT.1.OR.IOPC.GT.30)GO TO 150

WRITE(6,20)IOPC

IF(IREL.EQ.1)WRITE(6,20)IOPC

20 FORMAT(1H1, //5X, '*** OPCAO LIDA =', I4)

CALL MSGOP(IOPC,IREL)

CALL PAUSA

753 WRITE(6,25)

CALL MSG(607,IREL)

CALL SINO(K2,&3000,&752,&753)

752 IF(IXIST.EQ.1)GO TO 751

IF(IOPC.LE.5.AND.IOPC.NE.4)GO TO 751

CALL MSG(599,IREL)

CALL PAUSA

GO TO 3000

99 CALL MSG(620,IREL)

CALL PAUSA

GO TO 3000

751 IF(IOPC.NE.4)GO TO 250

WRITE(6,25)

```
CALL MSG(603,IREL)
DO 450 L=1,3
READ(5,31)SE
31  FORMAT(5X,A4)
IF(SE.EQ.SENHA)GO TO 250
NERRO=NERRO+1
IF(L.EQ.3)GO TO 450
WRITE(6,25)
CALL MSG(604,IREL)
450  CONTINUE
CALL MSG(605,IREL)
CALL PAUSA
GO TO 3000
150  WRITE(6,34)IOPC
IF(IREL.EQ.1)WRITE(8,34)IOPC
34  FORMAT(/5X,'*** OPCAO LIDA =',I4,2X,'INVALIDA, TENTE NOVA
MENTE')
CALL PAUSA
NERRO=NERRO+1
GO TO 3000
250  GO TO(100,200,300,400,500,600,700,800,900,1000,1100,1200,1
300,140
*0,1500,1600,1700,1800,1900,2000,2100,2200,2300,2400,2500,2
600,270
*0,2800,2900,3000),IOPC
100  CALL AZAR
RETURN
200  CALL CATA
RETURN
300  CALL LEGR
RETURN
400  CALL GRAVA
RETURN
500  CALL LEAL
RETURN
600  CALL SUCE
RETURN
700  CALL TOTSU
RETURN
800  CALL IMPA
RETURN
900  CALL ADJA
RETURN
1100 CALL VALORA
RETURN
```

```
1200 CALL FLOYD1  
      RETURN  
1300 CALL DMCRON  
      RETURN  
1400 CALL FLOYD2  
      RETURN  
1500 CALL RADI  
      RETURN  
1600 CALL FCONE  
      RETURN  
1700 CALL RADIV  
      RETURN  
1800 CALL FLUXO  
      RETURN  
1900 CALL DKSTRA  
      RETURN  
2100 CALL KRUSKA  
      RETURN  
2200 CALL DISOR  
      RETURN  
2300 RETURN  
2400 RETURN  
2500 RETURN  
2600 RETURN  
2700 RETURN  
2800 RETURN  
2900 RETURN  
      END
```

```
C  
C      GERA UM GRAFO ALEATORIAMENTE  
C
```

```
      SUBROUTINE AZAR  
      DIMENSION MADJ(40,40),NSUC(40)  
      COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC  
      COMMON /BLK2/IOB,NVER,NARC,MADJ,NSUC  
24     FORMAT(1H1,////////////////////////////////////)  
25     FORMAT(1H1,////////////////////////////////////)  
751    CALL INFO  
401    WRITE(6,24)  
      CALL MSG(606,IREL)  
      CALL LINHA(4)  
      READ(5,10,DATA=99)NARC  
10     FORMAT(I4)  
      CALL ZERO(NARC,2401)  
      WRITE(6,50)NARC
```

```
50  FORMAT(/5X,'*** VALOR LIDO =',I5)
    IF(IREL.EQ.1)WRITE(8,50)NARC
    NAR2=NVER*(NVER-1)
    IF(IOR.EQ.0)NAR2=NAR2/2
    IF(NARC.LE.NAR2)GO TO 444
    WRITE(6,61)NAR2
61  FORMAT(/5X,'*** NUMERO MAXIMO DE ARCOS/ARESTAS PARA SEU G
    RAFO E',
    *I6,/9X,'ENTRE NOVAMENTE')
    IF(IREL.EQ.1)WRITE(8,61)NAR2
    NERRO=NERRO+1
    CALL PAUSA
    GO TO 401
99  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 401
444  CALL PAUSA
    ENTRY ALEA(R,PROB)
666  WRITE(6,25)
    CALL MSG(608,IREL)
    CALL LINHA(5)
    READ(5,20,DATA=999)R
20  FORMAT(F6.2)
    WRITE(6,51)R
    IF(IREL.EQ.1)WRITE(8,51)R
51  FORMAT(/5X,'*** VALOR LIDO =',F13.2)
    CALL PAUSA
    IF(NARC.EQ.0.OR.IGPC.EQ.5)GO TO 333
    PROB=FLOAT(NARC)/FLOAT(NAR2)
    GO TO 555
999  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 666
333  WRITE(6,25)
    CALL MSG(609,IREL)
    CALL LINHA(3)
    READ(5,30,DATA=88)JPROB
    PROB=FLOAT(JPROB/100.)
30  FORMAT(I3)
    IF(PROB.GT.0.0.AND.PROB.LE.1.0)GO TO 222
    NERRO=NERRO +1
    WRITE(6,44)PROB
    IF(IREL.EQ.1)WRITE(8,44)PROB
```

```
44  FORMAT(/5X,'*** PROBABILIDADE LIDA =', F6.2,2X,'INVALIDA,
    TENTE N
*OVAMENTE!')
    CALL PAUSA
    GO TO 333
88  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 333
222 WRITE(6,91)PROB
    IF(IREL.EQ.1)WRITE(8,91)PROB
91  FORMAT(/5X,'*** PROBABILIDADE LIDA =', F6.2)
    CALL PAUSA
    IF(IOPC.EQ.5)RETURN
555 K=0
    IARC=0
    IF(IOR.EQ.0)GO TO 600
    DO 150 I=1,NVER
    DO 150 J=1,NVER
    IF(I.EQ.J)GO TO 150
22  X=RANDOM(R)
    IF(X.EQ.0.)GO TO 22
    IF(X.GT.PROB)GO TO 150
    NSUC(I)=NSUC(I)+1
    MADJ(I,J)=1
    IARC=IARC+1
    IF(NARC.EQ.0)GO TO 150
    K=K+1
    IF(NARC.LE.K)GO TO 350
150 CONTINUE
    GO TO 350
600 DO 250 I=1,NVER-1
    DO 250 J=I+1,NVER
33  X=RANDOM(R)
    IF(X.EQ.0.)GO TO 33
    IF(X.GT.PROB)GO TO 250
    NSUC(I)=NSUC(I)+1
    NSUC(J)=NSUC(J)+1
    MADJ(I,J)=1
    MADJ(J,I)=1
    IARC=IARC+1
    IF(NARC.EQ.0)GO TO 250
    K=K+1
    IF(NARC.LE.K)GO TO 350
250 CONTINUE
```

```
350 NARC=IARC
    WRITE(6,93)NARC,NVER
    IF(IREL.EQ.1)WRITE(8,93)NARC,NVER
93  FORMAT(1H1, //5X, '*** O GRAFO GERADO TEM :', //9X, I6, 2X, 'ARC
    OS/AREST
    *AS', //9X, I6, 2X, 'VERTICES')
    IF(IOR)521,522,521
521  CALL MSG(610,IREL)
    GO TO 523
522  CALL MSG(611,IREL)
523  CALL PAUSA
    IXIST=1
650  WRITE(6,25)
    CALL MSG(612,IREL)
    CALL SINO(KK,8751,8752,8650)
752  CALL MENSA
    IF(IOPC.EQ.1)CALL NEWOP
    RETURN
    END
```

```
C
C  LE UM GRAFO FORNECIDO PELO TERMINAL
C
```

```
    SUBROUTINE LEGR
    INTEGER SUÇ(40)
    DIMENSION MADJ(40,40),NSUC(40)
    COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
    COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
25  FORMAT(1H1, //5X, '*****')
751  CALL INFO
    I=1
201  WRITE(6,25)
    WRITE(6,10)I
    IF(IREL.EQ.1)WRITE(6,10)I
10  FORMAT(/5X, '*** ENTRE COM O SEMIGRAU EXTERIOR DO VERTI
    CE', I3)
    CALL LINHA(?)
    READ(5,11,DATA=99)NSUC(I)
11  FORMAT(I2)
    CALL ZERO(NSUC(I),8201)
    IF(NSUC(I).NE.0)GO TO 203
    WRITE(6,31)I
    IF(IREL.EQ.1)WRITE(6,31)I
31  FORMAT(/5X, '*** VERTICE', I3, 2X, 'NAO TEM SUCESSORES')
    CALL PAUSA
208  WRITE(6,25)
```

```
      CALL MSG(614,IREL)
      CALL SIND(II,8201,8206,8208)
206  IF(I.GE.NVER)GO TO 601
      I=I+1
      GO TO 201
99   CALL MSG(620,IREL)
      CALL PAUSA
      NERRO=NERRO+1
      GO TO 201
203  IF(NSUC(I).LE.NVER)GO TO 204
      NERRO=NERRO +1
      WRITE(6,27)NSUC(I)
      IF(IREL.EQ.1)WRITE(8,27)NSUC(I)
27   FORMAT(/5X,'*** VALOR LIDO =',I5,2X,'MAIOR QUE NUMERO DE
      VERTICES
      *, -TENETE NOVAMENTE')
      CALL PAUSA
      GO TO 201
204  WRITE(6,25)
      WRITE(6,26)NSUC(I),I
      IF(IREL.EQ.1)WRITE(8,26)NSUC(I),I
26   FORMAT(/5X,'*** ENTRE COM OS',I4,2X,'SUCESSORES DO VERTIC
      E',I4)
207  KS=NSUC(I)
      CALL LER(1,KS)
      READ(5,20,DATA=999)(SUC(K),K=1,KS)
20   FORMAT(40I2)
      DO 400 K=1,NSUC(I)
      IF(SUC(K).LE.NVER)GO TO 400
      NERRO=NERRO +1
      WRITE(6,27)SUC(K)
      IF(IREL.EQ.1)WRITE(8,27)SUC(K)
      CALL PAUSA
      GO TO 204
999  CALL MSG(620,IREL)
      CALL PAUSA
      NERRO=NERRO+1
      GO TO 204
400  CALL NEG(SUC(K),8204)
      GO TO 100
      CALL PAUSA
      NERRO=NERRO +1
      GO TO 204
100  WRITE(6,45)I,NSUC(I),(SUC(K),K=1,NSUC(I))
      IF(IREL.EQ.1)WRITE(8,45)I,NSUC(I),(SUC(K),K=1,NSUC(I))
```

```
45  FORMAT(//5X,'*** VALORES LIDOS = VERTICE',I3,2X,' SEMIGRAU
    EXTERI
*OR ',I4,/,9X,'SUCESSORES' , 4(T20,10I5,//))
    CALL PAUSA
101  WRITE(6,25)
    CALL MSG(614,IREL)
    CALL SINO(JJ,8201,8600,8101)
600  NARC=NARC+NSUC(I)
    DO 300 K=1,NSUC(I)
    IF(MADJ(I,SUC(K)).NE.0)GO TO 301
    MADJ(I,SUC(K))=1
    GO TO 300
301  WRITE(6,94)SUC(K),I
    IF(IREL.EQ.1)WRITE(8,94)SUC(K),I
94   FORMAT(//5X,'*** VERTICE',I3,2X,'REPETIDO COMO SUCESSOR DE
    ',I3,2X,
* 'TENTE NOVAMENTE')
    CALL PAUSA
    NARC=NARC-NSUC(I)
    DO 311 K1=1,40
    MADJ(I,K1)=0
311  CONTINUE
    NERRO=NERRO+1
    GO TO 201
300  CONTINUE
    IF(I.GE.NVER)GO TO 601
    I=I+1
    GO TO 201
601  IF(IOR.EQ.1)GO TO 603
    DO 500 I=1,NVER
    DO 500 J=1,I
500  MADJ(J,I)=MADJ(I,J)
    NARC=0
    DO 501 I=1,NVER
    NSUC(I)=0
    DO 501 J=1,NVER
    NSUC(I)=NSUC(I)+MADJ(I,J)
501  NARC=NARC+MADJ(I,J)
    NARC=NARC/2
603  WRITE(6,93)NARC,NVER
    IF(IREL.EQ.1)WRITE(8,93)NARC,NVER
93   FORMAT(1H1,///5X,'*** O GRAFO LIDO TEM :',//9X,I6,2X,'ARCO
    S/ARESTA
*S',//9X,I6,2X,'VERTICES')
    IF(IOR)521,522,521
```



```
521 CALL MSG(610,IREL)
    GO TO 523
522 CALL MSG(611,IREL)
523 CALL PAUSA
    IXIST=1
650 WRITE(6,25)
    CALL MSG(615,IREL)
    CALL SINO(KK,&751,&752,&650)
752 CALL MENSA
    IF(IOPC.EQ.5)RETURN
    IF(IOPC.EQ.3)CALL NEWOP
    RETURN
    END
```

```
C
C  MOSTRA NA TELA A MATRIZ DE ADJACENCIA, PARA GRAFOS ATE 15
    VERTICES
```

```
C
    SUBROUTINE ADJA
    DIMENSION MADJ(40,40),NSUC(40)
    COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
    COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
    IF(NVER.GT.15)GO TO 100
    WRITE(6,10)NVER,NVER,(K,K=1,NVER)
    IF(IREL.EQ.1)WRITE(8,10)NVER,NVER,(K,K=1,NVER)
10  FORMAT(IH1,I1X,'*** MATRIZ DE ADJACENCIA DO GRAFO ',I3,I1X,
    'X',I3,/)
    *//10X,I5I3,//)
    DO 200 K=1,NVER
    WRITE(6,20)K,(MADJ(K,I),I=1,NVER)
    IF(IREL.EQ.1)WRITE(8,20)K,(MADJ(K,I),I=1,NVER)
20  FORMAT(/5X,I3,2X,I5I3)
200  CONTINUE
    CALL PAUSA
    GO TO 300
100  WRITE(6,30)NVER
    IF(IREL.EQ.1)WRITE(8,30)NVER
30  FORMAT(/5X,'*** NUMERO DE VERTICES =',I4,2X,'MATRIZ NAO P
    ODE SER
    *OBSERVADA NA TELA')
    CALL PAUSA
    NERRO=NERRO +1
300  IF(IOPC.EQ.9)CALL NEWOP
    RETURN
    END
```

```
C
```

C MOSTRA NA TELA OS SUCESSORES DE TODOS OS VERTICES

```
C
SUBROUTINE SUCE
INTEGER SUC(40)
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
I=1
550 CALL MSG(616,IREL)
DO 450 KK=1,3
IF(NSUC(I).EQ.0)GO TO 250
K=0
DO 100 J=1,NVER
IF(MADJ(I,J).EQ.0)GO TO 100
K=K+1
SUC(K)=J
100 CONTINUE
WRITE(6,10)I,NSUC(I),(SUC(L),L=1,NSUC(I))
IF(IREL.EQ.1)WRITE(8,10)I,NSUC(I),(SUC(L),L=1,NSUC(I))
10 FORMAT(/5X,'*** VERTICE',I3,2X,' SEMIGRAU EXTERIOR ',I4, /
9X,'SUCE
*SSORES' , 4(T20,10I5,//))
GO TO 150
250 WRITE(6,20)I
IF(IREL.EQ.1)WRITE(8,20)I
20 FORMAT(/5X,'*** VERTICE',I3,2X,'NAD TEM SUCESSORES')
150 IF(I.EQ.NVER)GO TO 650
I=I+1
450 CONTINUE
CALL PAUSA
GO TO 550
650 CALL PAUSA
IF(IOPC.EQ.6)CALL NEWOP
RETURN
END
```

C MOSTRA NA TELA O SEMIGRAU EXTERIOR DE CADA VERTICE

```
C
SUBROUTINE TOTSU
DIMENSION IVER(4),NSAI(4)
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
DO 200 M=1,NVER
NSUC(M)=0
```

```
DO 200 N=1,NVER
NSUC(M)=NSUC(M)+MADJ(M,N)
200 CONTINUE
CALL MSG(617,IREL)
K=NVER/4
N=4*K
IF(N.NE.NVER)K=K+1
DO 100 I=1,K
KK=1
DO 150 J=I,I+3*K,K
IF(J.GT.NVER)GO TO 150
IVER(KK)=J
NSAI(KK)=NSUC(J)
KK=KK+1
150 CONTINUE
WRITE(6,10)(IVER(L),NSAI(L),L=1,KK=1)
IF(IREL.EQ.1)WRITE(8,10)(IVER(L),NSAI(L),L=1,KK=1)
10 FORMAT(/5X,4(I3,4X,I3,6X))
100 CONTINUE
CALL PAUSA
IF(IOPC.EQ.7)CALL NEWOP
RETURN
END
```

```
C
C IMPRIME A MATRIZ DE ADJACENCIA
C
```

```
SUBROUTINE IMPA
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
WRITE(8,10)NVER,NVER,(K,K=1,NVER)
10 FORMAT(1H1,///35X,'MATRIZ DE ADJACENCIA DO GRAFO ',I3,1X,'X',
,I3,///1
*0X,40I3)
DO 100 K=1,NVER
WRITE(8,20)K,(MADJ(K,J),J=1,NVER)
20 FORMAT(/5X,I3,2X,40I3)
100 CONTINUE
CALL MSG(618,IREL)
CALL PAUSA
IF(IOPC.EQ.8)CALL NEWOP
RETURN
END
```

```
C
C PARA EXECUTAR OUTRA OPCAO
```

```
C
SUBROUTINE NEWOP
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
40 WRITE(6,25)
25 FORMAT(1H1,////////////////////////////////////)
CALL MSG(619,IREL)
CALL SINO(KK,820,830,840)
20 CALL DPCAO
30 RETURN
END

C
C ANALISE DO SIM OU NAO
C
SUBROUTINE SINO(ILR,*,*,*)
DIMENSION LER(03),ASK(2)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
DATA ASK/'NAO','SIM'/
DATA BRC,SIM,NAO/' ','S','N'/
CALL LINHA(3)
READ(5,10)LER
10 FORMAT(3A1)
DO 100 I=1,3
IF(LER(I).IS.BRC)GO TO 100
IF(LER(I).IS.SIM)GO TO 600
IF(LER(I).IS.NAO)GO TO 700
GO TO 500
100 CONTINUE
GO TO 500
600 ILR=1
WRITE(6,37)ASK(ILR+1)
IF(IREL.EQ.1)WRITE(8,37)ASK(ILR+1)
37 FORMAT(/SX,'*** INFORMACAO LIDA =',2X,A3)
CALL PAUSA
RETURN 1
700 ILR=0
WRITE(6,37)ASK(ILR+1)
IF(IREL.EQ.1)WRITE(8,37)ASK(ILR+1)
CALL PAUSA
RETURN 2
500 ILR=2
NERRO=NERRO +1
CALL MSG(620,IREL)
CALL PAUSA
```

```
RETURN 3  
END
```

```
C  
C REJEITA VALORES MENORES OU IGUAIS A ZERO  
C  
SUBROUTINE NEG(IVAL,*)  
REAL IVAL  
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC  
IF(IVAL.GT.0)RETURN  
NERRO=NERRO +1  
WRITE(6,10)IVAL  
IF(IREL.EQ.1)WRITE(8,10)IVAL  
10 FORMAT(/5X,'*** VALOR LIDO =',F6.2,2X,'INVALIDO, TENTE NO  
VAMENTE'  
*)  
CALL PAUSA  
RETURN 1  
END
```

```
C  
C REJEITA SOMENTE VALORES MENORES QUE ZERO  
C  
SUBROUTINE ZERO(IVAL,*)  
REAL IVAL  
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC  
IF(IVAL.GE.0)RETURN  
NERRO=NERRO +1  
WRITE(6,10)IVAL  
IF(IREL.EQ.1)WRITE(8,10)IVAL  
10 FORMAT(/5X,'*** VALOR LIDO =',F6.2,2X,'INVALIDO, TENTE NO  
VAMENTE'  
*)  
CALL PAUSA  
RETURN 1  
END
```

```
C  
C PERMITE ARMAZENAR UM GRAFO EM ARGGRA  
C  
SUBROUTINE GRAVA  
INTEGER TOP  
DIMENSION MADJ(40,40),NSUC(40)  
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC  
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC  
COMMON /BLK3/TOP,MAX  
READ(15,1)TOP  
READ(15,2)LAST
```

```
ITOP=TOP
NAST=LAST
IF(TOP.EQ.0)TOP=1
LAST=LAST+1
CALL PUSH(LAST,&100)
IF(TOP=2)250,150,250
150 CALL PUSH(-1,&100)
CALL PUSH(5,&100)
CALL PUSH(1,&100)
GO TO 250
100 TOP=ITOP
WRITE(6,10)NAST
IF(IREL.EQ.1)WRITE(8,10)NAST
10 FORMAT(1H1,///5X,'*** EXCEDEU A CAPACIDADE RESERVADA, O GR
AFQ NAO
*FOI ARMAZENADO',///5X,'*** ULTIMO GRAFO ARMAZENADO NUMERO'
,I5)
GO TO 350
250 NEXT=TOP+1
TOP=TOP+1
I=1
550 IF(NSUC(I).EQ.0)GO TO 450
CALL PUSH(I,&100)
CALL PUSH(NSUC(I),&100)
DO 300 J=1,NVER
IF(MADJ(I,J).EQ.0)GO TO 300
CALL PUSH(J,&100)
300 CONTINUE
450 I=I+1
IF(I.LE.NVER)GO TO 550
WRITE(15,1)TOP
WRITE(15,2)LAST
NTOP=TOP+1
WRITE(15,NEXT)NTOP
WRITE(6,20)LAST
IF(IREL.EQ.1)WRITE(8,20)LAST
20 FORMAT(1H1,///5X,'*** O GRAFO ARMAZENADO RECEBEU O NUMERO'
,I5)
350 WRITE(6,25)TOP
IF(IREL.EQ.1)WRITE(8,25)TOP
25 FORMAT(///5X,'*** ULTIMA POSICAO GRAVADA',I7)
CALL PAUSA
IF(IOPC.EQ.4)CALL NEWDP
RETURN
END
```

```
C
C
C
LE UM GRAFO ARMazenado EM ARGRA

SUBROUTINE CATA
INTEGER TOP
DIMENSION VAL(40,40)
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK3/TOP,MAX
COMMON /BLK4/IVLR,VAL
READ(15,1)TOP
IF(TOP.NE.0)GO TO 851
CALL MSG(653,IREL)
CALL PAUSA
GO TO 853
851 DO 100 L=1,40
    NSUC(L)=0
    DO 100 J=1,40
        MADJ(L,J)=0
100 CONTINUE
    IOR=1
    KK=0
    IVLR=0
    NARC=0
    NVER=0
550 WRITE(6,25)
25  FORMAT(1H1,////////////////////////////////////)
    CALL MSG(621,IREL)
    CALL LINHA(3)
    READ(5,10,DATA=99)NGRA
10  FORMAT(I3)
    CALL NEG(NGRA,8550)
    WRITE(6,23)NGRA
23  IF(IREL.EQ.1)WRITE(8,23)NGRA
    FORMAT(/5X,'*** NUMERO DO GRAFO A SER USADO =',I5)
    CALL PAUSA
    READ(15,2)LAST
    READ(15,3)IGRA
    READ(15,4)NEXT
    K=3
450 IF(NGRA.EQ.IGRA)GO TO 250
    IF(IGRA.EQ.LAST)GO TO 350
    READ(15,NEXT)IGRA
    K=NEXT
```

```
READ(15,'NEXT+1')NEXT
GO TO 450
99 CALL MSG(620,IREL)
CALL PAUSA
NERRO=NERRO+1
GO TO 550
350 KK=KK+1
IF(KK.GT.3)GO TO 650
WRITE(6,20)NGRA
IF(IREL.EQ.1)WRITE(8,20)NGRA
20 FORMAT(/5X,'*** GRAFO',I5,2X,'NAO EXISTE, TENTE NOVAMENTE
')
CALL PAUSA
GO TO 550
650 CALL MSG(605,IREL)
CALL PAUSA
CALL OPCAO
RETURN
250 K=K+1
READ(15,'K')IFIM
IFIM=IFIM-1
750 CALL POP(I,K)
CALL POP(NSUC(I),K)
NARC=NARC+NSUC(I)
DO 101 L=1,NSUC(I)
CALL POP(J,K)
IF(J.GT.NVER)NVER=J
IF(I.GT.NVER)NVER=I
MADJ(I,J)=1
101 CONTINUE
IF(K.LT.IFIM)GO TO 750
WRITE(6,91)NARC,NVER
IF(IREL.EQ.1)WRITE(8,91)NARC,NVER
91 FORMAT(1H1,///5X,'*** O GRAFO CHAMADO TEM :',//9X,I6,2X,'A
RCOS/ARE
*STAS',//9X,I6,2X,'VERTICES')
CALL MSG(610,IREL)
CALL PAUSA
IXIST=1
752 WRITE(6,25)
CALL MSG(622,IREL)
CALL SING(II,&851,&852,&752)
852 CALL MENSA
853 IF(IOPC.EQ.2)CALL NEWOP
RETURN
```


END

C
C AUXILIAR PARA CHAMAR AS ROTINAS : ADJA, TOTSU E SUCE
C

```

SUBROUTINE MENSA
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
24  FORMAT(1H1,////////////////////////////////////)
25  FORMAT(1H1,////////////////////////////////////)
653  WRITE(6,24)
     CALL MSG(623,IREL)
     CALL SINO(II,8651,8652,8653)
651  CALL ADJA
652  WRITE(6,25)
     CALL MSG(624,IREL)
     CALL SINO(II,8551,8552,8652)
551  CALL TOTSU
552  WRITE(6,25)
     CALL MSG(625,IREL)
     CALL SINO(II,8751,8752,8552)
751  CALL SUCE
752  RETURN
END
```

C
C COLOCA UM ELEMENTO EM ARQGRA
C

```

SUBROUTINE PUSH(IX,*)
INTEGER TOP
COMMON /BLK3/TOP,MAX
TOP=TOP+1
IF(TOP.GT.MAX)GO TO 100
WRITE(15'TOP)IX
RETURN
100  TOP=TOP-1
     RETURN 1
END
```

C
C LE UM ELEMENTO EM ARQGRA
C

```

SUBROUTINE POP(IX,IPONT)
COMMON /BLK3/TOP,MAX
IPONT=IPONT+1
READ(15'IPONT)IX
RETURN
```

```
END
C
C PERMITE UMA PARADA DA TELA PARA OBSERVACAO
C
SUBROUTINE PAUSA
WRITE(6,10)
10  FORMAT(//1X,'==> XMT')
    READ(5)
    RETURN
    END
C
C AUXILIAR PARA LEITURA DO NUMERO DE VERTICES E ORIENTACAO
C
SUBROUTINE INFO
DIMENSION MADJ(40,40),NSUC(40)
DIMENSION VAL(40,40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK4/IVLR,VAL
DO 750 I=1,40
NSUC(I)=0
DO 750 J=1,40
MADJ(I,J)=0
750 CONTINUE
NVER=0
NARC=0
IVLR=0
100 WRITE(6,25)
25  FORMAT(1H1,////////////////////////////////////)
    CALL MSG(626,IREL)
    CALL SINO(IOR,&500,&500,&100)
500 WRITE(6,25)
    CALL MSG(627,IREL)
    CALL LINHA(2)
    READ(5,10,DATA=99)NVER
10  FORMAT(I2)
    CALL NEG(NVER,&500)
    IF(NVER.GT.40)GO TO 600
    WRITE(6,50)NVER
    IF(IREL.EQ.1)WRITE(8,50)NVER
50  FORMAT(//5X,'*** VALOR LIDO =',I5)
    CALL PAUSA
    RETURN
600 NERRO=NERRO+1
    WRITE(6,60)NVER
```

```
IF(IREL.EQ.1)WRITE(8,60)NVER
60  FORMAT(//5X,'*** VALOR LIDO =',I5,2X,'MAIOR QUE 40, TENTE
    NOVAMENT
    *E')
    CALL PAUSA
    GO TO 500
99  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 500
    END

C
C   LE UM GRAFO PELO TERMINAL E GERA OUTRA PARTE ALEATORIAMEN
C   TE

SUBROUTINE LEAL
DIMENSION MADJ(40,40),NSUC(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
750 CALL MSG(628,IREL)
    CALL PAUSA
    CALL LEGR
    CALL ALEA(R,PROB)
    DO 100 I=1,NVER
    DO 100 J=1,NVER
    IF(I.EQ.J)GO TO 100
    IF(MADJ(I,J).EQ.1)GO TO 100
22  X=RANDOM(R)
    IF(X.EQ.0.)GO TO 22
    IF(X.GT.PROB)GO TO 100
    MADJ(I,J)=1
    NSUC(I)=NSUC(I)+1
    NARC=NARC+1
    IF(IOR.EQ.1)GO TO 100
    MADJ(J,I)=1
    NSUC(J)=NSUC(J)+1
100 CONTINUE
    WRITE(6,20)NARC,NVER
    IF(IREL.EQ.1)WRITE(8,20)NARC,NVER
20  FORMAT(1H1,//5X,'*** O GRAFO FINAL TEM :',//9X,I6,2X,'ARC
    OS/AREST
    *AS',//9X,I6,2X,'VERTICES')
    IF(IOR)520,530,520
520 CALL MSG(610,IREL)
    GO TO 540
```

```
530 CALL MSG(611,IREL)
540 CALL PAUSA
770 WRITE(6,25)
25  FORMAT(1H1,////////////////////////////////////)
    CALL MSG(629,IREL)
    CALL SINO(KK,8750,8760,8770)
760 CALL MENSA
    IF(IOPC.EQ.5)CALL NEWOP
    RETURN
    END
```

```
C
C  PERMITE DAR VALORES AOS ARCOS/ARESTAS DE UM GRAFO
C
```

```
    SUBROUTINE VALORA
    DIMENSION MADJ(40,40),NSUC(40),VAL(40,40)
    DIMENSION ISAI(9),JSAI(9)
    COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
    COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
    COMMON /BLK4/IVLR,VAL
24  FORMAT(1H1,////////////////////////////////////)
25  FORMAT(1H1,////////////////////////////////////)
300 WRITE(6,25)
    CALL MSG(630,IREL)
    CALL SINO(KI,8100,8200,8300)
100 WRITE(6,25)
    CALL MSG(631,IREL)
    CALL LINHA(5)
    READ(5,10,DATA=99)CONST
10  FORMAT(F6.2)
    CALL NEG(CONST,8100)
    WRITE(6,12)CONST
    IF(IREL.EQ.1)WRITE(8,12)CONST
12  FORMAT(/5X,'*** VALOR LIDO =',F9.2)
    CALL PAUSA
    DO 400 I=1,NVER
    DO 400 J=1,NVER
    IF(I.EQ.J)GO TO 400
    IF(MADJ(I,J))20,20,21
20  VAL(I,J)=1E68
    GO TO 400
21  VAL(I,J)=CONST
400 CONTINUE
    GO TO 700
99  CALL MSG(620,IREL)
    CALL PAUSA
```

```

    NERRO=NERRO+1
    GO TO 100
999  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 501
200  K=0
    JFIM=NVER
    DO 500 I=1,NVER
    IF(IOR.EQ.0)JFIM=I
    DO 500 J=1,JFIM
    IF(I.EQ.J)GO TO 500
    IF(MADJ(I,J))30,30,31
30   VAL(I,J)=1E68
    GO TO 500
31   K=K+1
    ISAI(K)=I
    JSAI(K)=J
    IF(K.LE.8)GO TO 500
501  WRITE(6,24)
    WRITE(6,14)(ISAI(L),JSAI(L),L=1,K)
    IF(IREL.EQ.1)WRITE(8,14)(ISAI(L),JSAI(L),L=1,K)
14   FORMAT(/5X,'*** ENTRE COM OS VALORES DOS ARCOS/ARESTAS AB
    AIXO :',
    * /5X,9('(',I2,',',I2,',',I2,',')'))
    CALL LER(2,K)
    READ(5,40,DATA=999)(VAL(ISAI(L),JSAI(L)),L=1,K)
40   FORMAT(9F6.2)
    K=0
500  CONTINUE
    IF(K.NE.0)GO TO 501
    IF(IOR.EQ.1)GO TO 700
    DO 600 I=1,NVER
    DO 600 J=1,I
    VAL(J,I)=VAL(I,J)
600  CONTINUE
700  CALL MSG(632,IREL)
    IVLR=I
    CALL PAUSA
450  WRITE(6,25)
    CALL MSG(613,IREL)
    CALL SINO(K2,&300,&350,&450)
350  WRITE(6,25)
    CALL MSG(633,IREL)
    CALL SINO(K2,&150,&250,&350)

```

```
150 WRITE(8,16)NVER,NVER,(I,I=1,NVER)
16  FORMAT(1H1,///35X,'MATRIZ DE VALORES DO GRAFO',I3,1X,'
X',I3,//
*,4(11X,10(I2,10X),//)
DO 800 I=1,NVER
WRITE(8,17)I,(VAL(I,J),J=1,NVER)
17  FORMAT(/,I4,1X,4(10(F10.2,2X),/5X))
800  CONTINUE
CALL MSG(618,IREL)
CALL PAUSA
250 IF(IOPC.EQ.11)CALL NEWOP
101  RETURN
END
```

```
C
C  PERMITE ACHAR O CAMINHO MINIMO ENTRE DOIS VERTICES(FLOYD)
C
```

```
      SUBROUTINE FLOYD1
      DIMENSION MADJ(40,40),NSUC(40)
      DIMENSION VAL(40,40),DIST(40,40),MROT(40,40)
      DIMENSION LCAM(40)
      COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
      COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
      COMMON /BLK4/IVLR,VAL
      IF(IVLR.EQ.1)GO TO 100
25  FORMAT(1H1,//////////)
      CALL MSG(634,IREL)
      CALL PAUSA
      NERRO=NERRO+1
      CALL VALORA
100  DO 150 I=1,NVER
      DO 150 J=1,NVER
      IF(VAL(I,J)=1E67)140,151,151
140  MROT(I,J)=J
      GO TO 150
151  MROT(I,J)=0
150  DIST(I,J)=VAL(I,J)
      DO 200 K=1,NVER
      DO 200 I=1,NVER
      DO 200 J=1,NVER
      IF(DIST(I,K).GE.1E67.OR.DIST(K,J).GE.1E67)GO TO 200
      IF(DIST(I,J)=(DIST(I,K)+DIST(K,J)))200,200,201
201  DIST(I,J)=DIST(I,K)+DIST(K,J)
      MROT(I,J)=MROT(I,K)
200  CONTINUE
450  WRITE(6,25)
```

```
CALL MSG(635,IREL)
CALL SINO(K2,&250,&350,&450)
250 WRITE(8,12)NVER,NVER,(I,I=1,NVER)
12  FORMAT(1H1, //40X, 'MATRIZ DE DISTANCIA MINIMA DO GRAFO', I3,
      ' X', I3,
      *//,4(10X,10(I2,9X) ,/))
      DO 300 I=1,NVER
      WRITE(8,13)I,(DIST(I,J),J=1,NVER)
13  FORMAT(/I4,1X,4(10(F9.2,2X),)/5X))
300  CONTINUE
      CALL MSG(618,IREL)
      CALL PAUSA
350  WRITE(6,25)
      CALL MSG(636,IREL)
      CALL SINO(K2,&550,&650,&350)
550  WRITE(8,15)NVER,NVER,(I,I=1,NVER)
15  FORMAT(1H1, //30X, 'MATRIZ DE ROTEAMENTO MINIMO DO GRAFO',
      I3, ' X',
      *I3, //10X,40I3)
      DO 400 I=1,NVER
      WRITE(8,16)I,(MROT(I,J),J=1,NVER)
16  FORMAT(/5X, I3,2X,40I3)
400  CONTINUE
      CALL MSG(618,IREL)
      CALL PAUSA
650  WRITE(6,25)
      CALL MSG(637,IREL)
      CALL LINHA(6)
      READ(5,10,DATA=99)I,J
10  FORMAT(2I2)
      CALL NEG(I,&650)
      CALL NEG(J,&650)
      IF(I.LE.NVER.AND.J.LE.NVER)GO TO 750
      WRITE(6,17)I,J
17  IF(IREL.EQ.1)WRITE(8,17)I,J
      FORMAT(/5X, '*** VALORES LIDOS I=', I3,2X, 'J=', I3,2X, 'INCOM
      PATIVEL
      *COM NUMERO DE VERTICES',/9X, 'TENTE NOVAMENTE')
      NERRO=NERRO+1
      CALL PAUSA
      GO TO 650
99  CALL MSG(620,IREL)
      CALL PAUSA
      NERRO=NERRO+1
      GO TO 650
```

```
750 IF(DIST(I,J).GE.1E67)GO TO 950
    K=1
    LCAM(K)=I
851 KK=LCAM(K)
    IF(MROT(KK,J).EQ.J)GO TO 850
    K=K+1
    LCAM(K)=MROT(KK,J)
    GO TO 851
850 K=K+1
    LCAM(K)=J
    WRITE(6,18)I,J,(LCAM(M),M=1,K)
    IF(IREL.EQ.1)WRITE(8,18)I,J,(LCAM(M),M=1,K)
10 FORMAT(/5X,'*** VERTICES DO CAMINHO MINIMO ENTRE',2X,'( ',
    I2,' E ',
    *I3,' )',4(/9X,10I4))
    WRITE(6,19)I,J,DIST(I,J)
    IF(IREL.EQ.1)WRITE(8,19)I,J,DIST(I,J)
19 FORMAT(/5X,'*** DISTANCIA MINIMA ENTRE OS VERTICES',2X,'( ',
    I2,' E ',
    *I3,' )',2X,F9.2)
    CALL PAUSA
    GO TO 951
950 WRITE(6,20)I,J
    IF(IREL.EQ.1)WRITE(8,20)I,J
20 FORMAT(/5X,'*** NAO EXISTE NENHUM CAMINHO ENTRE OS VERTIC
    ES',2X,'
    *(',I2,' E ',I3,' )')
    CALL PAUSA
951 WRITE(6,25)
    CALL MSG(638,IREL)
    CALL SINO(K2,8650,8952,8951)
952 IF(IOPC.EQ.12)CALL NEWOP
    RETURN
    END

C
C DESCRICAO DE TODAS AS OPCOES
C
SUBROUTINE MSGOP(K,IREL)
GO TO(100,200,300,400,500,600,700,800,900,1000,1100,1200,1
300,1400
*,1500,1600,1700,1800,1900,2000,2100,2200,2300,2400,2500,26
00,2700,
*2800,2900,3000),K
100 WRITE(6,1)
    IF(IREL.EQ.1)WRITE(8,1)
```



```
RETURN
200 WRITE(6,2)
   IF(IREL.EQ.1)WRITE(8,2)
   RETURN
300 WRITE(6,3)
   IF(IREL.EQ.1)WRITE(8,3)
   RETURN
400 WRITE(6,4)
   IF(IREL.EQ.1)WRITE(8,4)
   RETURN
500 WRITE(6,5)
   IF(IREL.EQ.1)WRITE(8,5)
   RETURN
600 WRITE(6,6)
   IF(IREL.EQ.1)WRITE(8,6)
   RETURN
700 WRITE(6,7)
   IF(IREL.EQ.1)WRITE(8,7)
   RETURN
800 WRITE(6,8)
   IF(IREL.EQ.1)WRITE(8,8)
   RETURN
900 WRITE(6,9)
   IF(IREL.EQ.1)WRITE(8,9)
   RETURN
1000 WRITE(6,10)
   IF(IREL.EQ.1)WRITE(8,10)
   RETURN
1100 WRITE(6,11)
   IF(IREL.EQ.1)WRITE(8,11)
   RETURN
1200 WRITE(6,12)
   IF(IREL.EQ.1)WRITE(8,12)
   RETURN
1300 WRITE(6,13)
   IF(IREL.EQ.1)WRITE(8,13)
   RETURN
1400 WRITE(6,14)
   IF(IREL.EQ.1)WRITE(8,14)
   RETURN
1500 WRITE(6,15)
   IF(IREL.EQ.1)WRITE(8,15)
   RETURN
1600 WRITE(6,16)
   IF(IREL.EQ.1)WRITE(8,16)
```

```
RETURN
1700 WRITE(6,17)
    IF(IREL.EQ.1)WRITE(8,17)
    RETURN
1800 WRITE(6,18)
    IF(IREL.EQ.1)WRITE(8,18)
    RETURN
1900 WRITE(6,19)
    IF(IREL.EQ.1)WRITE(8,19)
    RETURN
2000 WRITE(6,20)
    IF(IREL.EQ.1)WRITE(8,20)
    RETURN
2100 WRITE(6,21)
    IF(IREL.EQ.1)WRITE(8,21)
    RETURN
2200 WRITE(6,22)
    IF(IREL.EQ.1)WRITE(8,22)
    RETURN
2300 WRITE(6,23)
    IF(IREL.EQ.1)WRITE(8,23)
    RETURN
2400 WRITE(6,24)
    IF(IREL.EQ.1)WRITE(8,24)
    RETURN
2500 WRITE(6,25)
    IF(IREL.EQ.1)WRITE(8,25)
    RETURN
2600 WRITE(6,26)
    IF(IREL.EQ.1)WRITE(8,26)
    RETURN
2700 WRITE(6,27)
    IF(IREL.EQ.1)WRITE(8,27)
    RETURN
2800 WRITE(6,28)
    IF(IREL.EQ.1)WRITE(8,28)
    RETURN
2900 WRITE(6,29)
    IF(IREL.EQ.1)WRITE(8,29)
    RETURN
3000 WRITE(6,30)
    IF(IREL.EQ.1)WRITE(8,30)
    RETURN
1  FORMAT(//15X,'GERA UM GRAFO ALEATORIAMENTE',//
    *5X,'PODE GERAR GRAFOS ORIENTADOS OU NAO',/
```

```
*5X, 'O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)',  
/  
*5X, 'A SEMENTE PARA GERACAO PODE SER QUALQUER NUMERO',/  
*5X, 'A PROBABILIDADE DEVE SER > ZERO E <=100',/  
*5X, 'PODE-SE LIMITAR O NUMERO DE ARCOS DESEJADOS',/  
*5X, 'ESTA E UMA ROTINA DE EDICAO DE UM GRAFO',/  
*5X, 'PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR',/  
*5X, 'APOS A GERACAO JA ESTAO A DISPOSICAO:',/  
*5X, '    A MATRIZ DE ADJACENCIA',/  
*5X, '    O SEMIGRAU EXTERIOR DE CADA VERTICE',/  
*5X, '    OS SUCESSORES DE CADA VERTICE')  
2  FORMAT(//15X, 'CHAMA UM GRAFO ARMAZENADO ANTERIORMENTE',//  
*5X, 'E NECESSARIO SABER O NUMERO DO GRAFO A SER USADO',/  
*5X, 'TODO GRAFO ARMAZENADO E ORIENTADO',/  
*5X, 'ESTA E UMA ROTINA DE EDICAO DE UM GRAFO',/  
*5X, 'PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR',/  
*5X, 'APOS A LEITURA JA ESTAO A DISPOSICAO:',/  
*5X, '    A MATRIZ DE ADJACENCIA',/  
*5X, '    O SEMIGRAU EXTERIOR DE CADA VERTICE',/  
*5X, '    OS SUCESSORES DE CADA VERTICE')  
3  FORMAT(//15X, 'LE UM GRAFO FORNECIDO PELO TERMINAL',//  
*5X, 'O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)',  
/  
*5X, 'NAO E ACONSELHAVEL O USO PARA GRAFOS GRANDES(>15 VERTI  
CES)',/  
*5X, 'PARA CADA VERTICE FORNECA TODOS OS SUCESSORES',/  
*5X, 'SE O VERTICE NAO TEM SUCESSORES ENTRE ZERO',/  
*5X, 'PODE-SE LER GRAFO ORIENTADO OU NAO',/  
*5X, 'PARA OS NAO ORIENTADOS PODE-SE ENTRAR SO COM OS SUCESS  
ORES',/  
*5X, '    DO TRIANGULO INFERIOR DA MATRIZ DE ADJACENCIA',/  
*5X, '    A MATRIZ E SIMETRIZADA EM RELACAO AO TRIANGULO INF  
ERIOR',/  
*5X, 'ESTA E UMA ROTINA DE EDICAO DE UM GRAFO',/  
*5X, 'PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR',/  
*5X, 'APOS A LEITURA JA ESTAO A DISPOSICAO:',/  
*5X, '    A MATRIZ DE ADJACENCIA',/  
*5X, '    O SEMIGRAU EXTERIOR DE CADA VERTICE',/  
*5X, '    OS SUCESSORES DE CADA VERTICE')  
4  FORMAT(//15X, 'ARMAZENA UM GRAFO PARA USO POSTERIOR',//  
*5X, 'PODE ARMAZENAR GRAFO LIDO, GERADO OU PARTE LIDO PARTE G  
ERADO',/  
*5X, 'E NECESSARIO CONHECER A SENHA AUTORIZADA',/  
*5X, 'A CADA GRAFO ARMAZENADO E DADO UM NUMERO SEQUENCIAL',/  
*5X, 'EXCEDENDO A CAPACIDADE DO ARQUIVO O GRAFO NAO E ARMAZE  
NADO',/
```

```
5 *5X,'AO FINAL E INDICADO O NUMERO QUE RECEBEU O SEU GRAFO')
  FORMAT(//15X,'LE UM GRAFO PELO TERMINAL DEPOIS GERA MAIS A
  RCOS',//
*5X,'O NUMERO MAXIMO DE VERTICES E 40 (DIFERENTE DE ZERO)',
/
*5X,'NAO E ACONSELHAVEL PARA GRAFOS COM MUITOS ARCOS',/
*5X,'INDIQUE PARA CADA VERTICE SE ELE TEM OU NAO SUCESSORES
',//
*5X,'PARA OS NAO ORIENTADOS PODE-SE ENTRAR SO COM OS SUCESS
ORES',//
*5X,' DO TRIANGULO INFERIOR DA MATRIZ DE ADJACENCIA',/
*5X,' A MATRIZ E SIMETRIZADA EM RELACAO AO TRIANGULO INF
ERIOR',/
*5X,'ESTA E UMA ROTINA DE EDICAO DE UM GRAFO',/
*5X,'PORTANTO DESTROI O GRAFO ANTERIOR SE EXISTIR',/
*5X,'APOS A GERACAO JA ESTAO A DISPOSICAO:',/
*5X,' A MATRIZ DE ADJACENCIA',/
*5X,' O SEMIGRAU EXTERIOR DE CADA VERTICE(TOTAL)',/
*5X,' OS SUCESSORES DE CADA VERTICE(TOTAL)')
6 FORMAT(//15X,'MOSTRA NA TELA OS SUCESSORES DE CADA VERTICE
',//
*5X,'SAO LISTADOS NA TELA TODOS OS VERTICES COM:',/
*5X,' O SEMIGRAU EXTERIOR ',/
*5X,' QUAIS SAO OS SUCESSORES DE CADA UM')
7 FORMAT(//15X,'MOSTRA O SEMIGRAU EXTERIOR DE CADA VERTIC
E',//
*5X,'SAO LISTADOS NA TELA TODOS OS VERTICES COM:',/
*5X,' O SEMIGRAU EXTERIOR DE CADA UM')
8 FORMAT(//15X,'SOLTA UMA LISTAGEM DA MATRIZ DE ADJACENCIA',
//
*5X,'IMPRIME A MATRIZ DE ADJACENCIA PARA QUALQUER GRAFO',/
*5X,'USADA QUANDO NAO SE PODE OBSERVAR A MATRIZ NA TELA',/
*5X,'PRINCIPALMENTE GRAFOS COM MAIS DE 15 VERTICES',/
*5X,'O ESCANINHO DA LISTAGEM TEM O NUMERO DO TERMINAL USADO
')
9 FORMAT(//15X,'MOSTRA NA TELA A MATRIZ DE ADJACENCIA',//
*5X,'SO PODE SER USADA PARA GRAFOS COM ATÉ 15 VERTICES',/
*5X,'PARA GRAFOS MAIORES USAR A OPCAO DE IMPRIMIR A MATRIZ'
)
10 FORMAT(//15X,'MOSTRA NA TELA A PROXIMA LISTA DE OPCOES',//
*5X,'A PROXIMA LISTA SERA A SEGUNDA',/
*5X,'PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA',/
*5X,'MESMO QUE NAO CONSTE DA LISTA APRESENTADA')
11 FORMAT(//15X,'PERMITE VALORAR UM GRAFO',//
*5X,'PODE-SE DAR UM VALOR CONSTANTE PARA TODOS ARCOS/ARESTA
S',/
```

```
*5X, 'GRAFOS NAO ORIENTADOS TERAQ OS VALORES SIMETRIZADOS', /
*5X, 'PODE-SE PEDIR UMA SAIDA DA MATRIZ DE VALORES', /
*5X, 'A IMPRESSAO CONTEM DUAS CASAS DECIMAIS', /
*5X, 'VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS(*)', /
*5X, 'O MAXIMO VALOR DEVERA SER MENOR QUE 10 ELEVADO A 5')
12  FORMAT(//15X, 'ALGORITMO DE FLOYD PARA CAMINHO MINIMO', //
*5X, 'SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS', /
*5X, 'CALCULA A MATRIZ DE DISTANCIA MINIMA DO GRAFO', /
*5X, 'CALCULA A MATRIZ DE ROTEAMENTO DO GRAFO', /
*5X, 'PODE-SE IMPRIMIR A MATRIZ DE DISTANCIA E DE ROTEAMENT
O ', /
*5X, 'PODE-SE PERGUNTAR A DISTANCIA MINIMA ENTRE DOIS VERTIC
ES', /
*5X, 'OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO
')
13  FORMAT(//15X, 'ALGORITMO DE DEMOUCCRON', //
*5X, 'DETERMINA OS NIVEIS DO GRAFO', /
*5X, 'INDICA A EXISTENCIA DE CIRCUITO', /
*5X, 'INDICA OS NIVEIS EXISTENTES ATE ENCONTRAR', /
*5X, 'O PRIMEIRO CIRCUITO', /
*5X, 'GRAFOS NAO ORIENTADOS, INDICA A EXISTENCIA DE CICLO')
14  FORMAT(//15X, 'ALGORITMO DE FLOYD PARA CAMINHO MAXIMO', //
*5X, 'SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS', /
*5X, 'O GRAFO NAO PODE TER NENHUM CIRCUITO', /
*5X, 'CALCULA O CAMINHO CRITICO NUMA REDE PERT', /
*5X, 'CALCULA A MATRIZ DE DISTANCIA MAXIMA DO GRAFO', /
*5X, 'CALCULA A MATRIZ DE ROTEAMENTO MAXIMO DO GRAFO', /
*5X, 'PODE-SE IMPRIMIR QUALQUER DAS MATRIZES', /
*5X, 'PODE-SE PEDIR A DISTANCIA MAXIMA ENTRE DOIS VERTICES',
/
*5X, 'OBTEM-SE O VALOR E OS VERTICES PERTENCENTES AO CAMINHO
')
15  FORMAT(//15X, 'RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAME
NTOS', //
*5X, 'VALIDA PARA GRAFOS NAO VALORADOS', /
*5X, 'CALCULA O RAIO E O DIAMETRO INTERIORES E EXTERIORES', /
*5X, 'DETERMINA O CENTRO E VERTICES PERIFERICOS', /
*5X, 'DETERMINA OS AFASTAMENTOS DOS VERTICES', /
*5X, 'CALCULA A MATRIZ DE DISTANCIAS', /
*5X, 'PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS', /
*5X, 'VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS(*)')
16  FORMAT(//15X, 'DECOMPOSICAO EM COMPONENTES F=CONEXAS (MALGRA
NGE)', //
*5X, 'DETERMINA AS COMPONENTES F=CONEXAS', /
*5X, 'INDICA SE O GRAFO E F=CONEXO', /
```

```
*5X, 'UTILIZA O ALGORITMO DE MALGRANGE',)
17  FORMAT(//15X, 'RAIO, DIAMETRO, CENTROS, PERIFERICOS E AFASTAME
    NTOS', //
    *5X, 'VALIDA PARA GRAFOS VALORADOS', /
    *5X, 'CALCULA O RAIO E O DIAMETRO INTERIORES E EXTERIORES', /
    *5X, 'DETERMINA OS CENTROS E VERTICES PERIFERICOS', /
    *5X, 'DETERMINA OS AFASTAMENTOS DOS VERTICES', /
    *5X, 'CALCULA A MATRIZ DE DISTANCIAS', /
    *5X, 'PODE-SE IMPRIMIR A MATRIZ E AFASTAMENTOS', /
    *5X, 'VALORES INFINITOS SERAO IMPRESSOS COM ASTERISCOS(*)', /
    *5X, 'PODE-SE DAR PESO AOS VERTICES')
18  FORMAT(//15X, 'ALGORITMO DE FORD-FULKERSON (FLUXO MAXIMO)', /
    /
    *5X, 'VALIDA PARA GRAFOS VALORADOS', /
    *5X, 'INDICA O VALOR DO FLUXO MAXIMO', /
    *5X, 'IMPRIME OS ARCOS PARA O FLUXO MAXIMO', /
    *5X, 'SAO INDICADOS OS ARCOS SATURADOS', /
    *5X, 'O FLUXO E DETERMINADO ENTRE O VERTICE 1 E NVER')
19  FORMAT(//15X, 'ALGORITMO DE DIJKSTRA (CAMINHO MINIMO)', //
    *5X, 'SO E PERMITIDO O USO PARA GRAFOS JA VALORADOS', /
    *5X, 'DEVE-SE INDICAR O VERTICE RAIZ', /
    *5X, 'FORNECE A DISTANCIA DA RAIZ A TODOS OS VERTICES', /
    *5X, 'PODE-SE ESCOLHER QUALQUER VERTICE PARA RAIZ', /
    *5X, 'USAR FLOYD PARA SABER OS VERTICES DO CAMINHO')
20  FORMAT(//15X, 'MOSTRA NA TELA A PROXIMA LISTA DE OPCOES', //
    *5X, 'A PROXIMA LISTA SERA A TERCEIRA', /
    *5X, 'PODE SER ESCOLHIDA QUALQUER OPCAO VALIDA', /
    *5X, 'MESMO QUE NAO CONSTE DA LISTA APRESENTADA', /
    *5X, 'DEVE ESCOLHER UMA OPCAO NA PROXIMA LISTA POIS SERA A U
    LTIMA')
21  FORMAT(//15X, 'ALGORITMO DE KRUSKAL (ARVORE PARCIAL MINIMA)
    ', //
    *5X, 'VALIDA SO PARA GRAFOS NAO ORIENTADOS E VALORADOS', /
    *5X, 'FORNECE OS ARCOS DA ARVORE PARCIAL')
22  FORMAT(//15X, 'DESORIENTAR UM GRAFO', //
    *5X, 'ALTERA O GRAFO ORIGINAL', /
    *5X, 'O GRAFO PASSA A SER NAO ORIENTADO', /
    *5X, 'SE HOUVER O ARCO (I,J) HAVERA O ARCO (J,I)', /
    *5X, 'NOS GRAFOS VALORADOS VAL(I,J) = VAL(J,I)', /
    *5X, 'PARA VALORES NOS DOIS SENTIDOS TEM-SE :', /
    *5X, '  VAL(I,J) = MAX(VAL(I,J) , VAL(J,I))')
23  FORMAT(//15X, 'LIVRE')
24  FORMAT(//15X, 'LIVRE')
25  FORMAT(//15X, 'LIVRE')
26  FORMAT(//15X, 'LIVRE')
```

```
27  FORMAT(//15X,'LIVRE')
28  FORMAT(//15X,'LIVRE')
29  FORMAT(//15X,'LIVRE')
30  FORMAT(//15X,'MOSTRA NA TELA A PROXIMA LISTA DE OPCOES',//
    *5X,'RETORNA A PRIMEIRA LISTA',/
    *5X,'JA FORAM APRESENTADAS TODAS OPCOES EXISTENTES!')
    END
```

```
C
C   CONTEM TODAS MENSAGENS DAS SUBROTINAS
C
```

```
    SUBROUTINE MSG(K,IREL)
    KK=K-590
    GO TO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,2
    1,22,23,
    *24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,4
    3,44,45,
    *46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,6
    5,66,67,
    *68,69),KK
1    WRITE(6,591)
    IF(IREL.EQ.1)WRITE(8,591)
    RETURN
2    WRITE(6,592)
    IF(IREL.EQ.1)WRITE(8,592)
    RETURN
3    WRITE(6,593)
    IF(IREL.EQ.1)WRITE(8,593)
    RETURN
4    RETURN
5    RETURN
6    RETURN
7    RETURN
8    RETURN
9    WRITE(6,599)
    IF(IREL.EQ.1)WRITE(8,599)
    RETURN
10   WRITE(6,600)
    IF(IREL.EQ.1)WRITE(8,600)
    RETURN
11   WRITE(6,601)
    IF(IREL.EQ.1)WRITE(8,601)
    RETURN
12   WRITE(6,602)
    IF(IREL.EQ.1)WRITE(8,602)
    RETURN
```

```
13  WRITE(6,603)
    IF(IREL.EQ.1)WRITE(8,603)
    RETURN
14  WRITE(6,604)
    IF(IREL.EQ.1)WRITE(8,604)
    RETURN
15  WRITE(6,605)
    IF(IREL.EQ.1)WRITE(8,605)
    RETURN
16  WRITE(6,606)
    IF(IREL.EQ.1)WRITE(8,606)
    RETURN
17  WRITE(6,607)
    IF(IREL.EQ.1)WRITE(8,607)
    RETURN
18  WRITE(6,608)
    IF(IREL.EQ.1)WRITE(8,608)
    RETURN
19  WRITE(6,609)
    IF(IREL.EQ.1)WRITE(8,609)
    RETURN
20  WRITE(6,610)
    IF(IREL.EQ.1)WRITE(8,610)
    RETURN
21  WRITE(6,611)
    IF(IREL.EQ.1)WRITE(8,611)
    RETURN
22  WRITE(6,612)
    IF(IREL.EQ.1)WRITE(8,612)
    RETURN
23  WRITE(6,613)
    IF(IREL.EQ.1)WRITE(8,613)
    RETURN
24  WRITE(6,614)
    IF(IREL.EQ.1)WRITE(8,614)
    RETURN
25  WRITE(6,615)
    IF(IREL.EQ.1)WRITE(8,615)
    RETURN
26  WRITE(6,616)
    IF(IREL.EQ.1)WRITE(8,616)
    RETURN
27  WRITE(6,617)
    IF(IREL.EQ.1)WRITE(8,617)
    RETURN
```



```
28  WRITE(6,618)
    IF(IREL.EQ.1)WRITE(8,618)
    RETURN
29  WRITE(6,619)
    IF(IREL.EQ.1)WRITE(8,619)
    RETURN
30  WRITE(6,620)
    IF(IREL.EQ.1)WRITE(8,620)
    RETURN
31  WRITE(6,621)
    IF(IREL.EQ.1)WRITE(8,621)
    RETURN
32  WRITE(6,622)
    IF(IREL.EQ.1)WRITE(8,622)
    RETURN
33  WRITE(6,623)
    IF(IREL.EQ.1)WRITE(8,623)
    RETURN
34  WRITE(6,624)
    IF(IREL.EQ.1)WRITE(8,624)
    RETURN
35  WRITE(6,625)
    IF(IREL.EQ.1)WRITE(8,625)
    RETURN
36  WRITE(6,626)
    IF(IREL.EQ.1)WRITE(8,626)
    RETURN
37  WRITE(6,627)
    IF(IREL.EQ.1)WRITE(8,627)
    RETURN
38  WRITE(6,628)
    IF(IREL.EQ.1)WRITE(8,628)
    RETURN
39  WRITE(6,629)
    IF(IREL.EQ.1)WRITE(8,629)
    RETURN
40  WRITE(6,630)
    IF(IREL.EQ.1)WRITE(8,630)
    RETURN
41  WRITE(6,631)
    IF(IREL.EQ.1)WRITE(8,631)
    RETURN
42  WRITE(6,632)
    IF(IREL.EQ.1)WRITE(8,632)
    RETURN
```

```
43  WRITE(6,633)
    IF(IREL.EQ.1)WRITE(8,633)
    RETURN
44  WRITE(6,634)
    IF(IREL.EQ.1)WRITE(8,634)
    RETURN
45  WRITE(6,635)
    IF(IREL.EQ.1)WRITE(8,635)
    RETURN
46  WRITE(6,636)
    IF(IREL.EQ.1)WRITE(8,636)
    RETURN
47  WRITE(6,637)
    IF(IREL.EQ.1)WRITE(8,637)
    RETURN
48  WRITE(6,638)
    IF(IREL.EQ.1)WRITE(8,638)
    RETURN
49  WRITE(6,639)
    IF(IREL.EQ.1)WRITE(8,639)
    RETURN
50  WRITE(6,640)
    IF(IREL.EQ.1)WRITE(8,640)
    RETURN
51  WRITE(6,641)
    IF(IREL.EQ.1)WRITE(8,641)
    RETURN
52  WRITE(6,642)
    IF(IREL.EQ.1)WRITE(8,642)
    RETURN
53  WRITE(6,643)
    IF(IREL.EQ.1)WRITE(8,643)
    RETURN
54  WRITE(6,644)
    IF(IREL.EQ.1)WRITE(8,644)
    RETURN
55  WRITE(6,645)
    IF(IREL.EQ.1)WRITE(8,645)
    RETURN
56  WRITE(6,646)
    IF(IREL.EQ.1)WRITE(8,646)
    RETURN
57  WRITE(6,647)
    IF(IREL.EQ.1)WRITE(8,647)
    RETURN
```

```
58  WRITE(6,648)
    IF(IREL.EQ.1)WRITE(8,648)
    RETURN
59  WRITE(6,649)
    IF(IREL.EQ.1)WRITE(8,649)
    RETURN
60  WRITE(6,650)
    IF(IREL.EQ.1)WRITE(8,650)
    RETURN
61  WRITE(6,651)
    IF(IREL.EQ.1)WRITE(8,651)
    RETURN
62  WRITE(6,652)
    IF(IREL.EQ.1)WRITE(8,652)
    RETURN
63  WRITE(6,653)
    IF(IREL.EQ.1)WRITE(8,653)
    RETURN
64  WRITE(6,654)
    IF(IREL.EQ.1)WRITE(8,654)
    RETURN
65  WRITE(6,655)
    IF(IREL.EQ.1)WRITE(8,655)
    RETURN
66  WRITE(6,656)
    IF(IREL.EQ.1)WRITE(8,656)
    RETURN
67  WRITE(6,657)
    IF(IREL.EQ.1)WRITE(8,657)
    RETURN
68  WRITE(6,658)
    IF(IREL.EQ.1)WRITE(8,658)
    RETURN
69  WRITE(6,659)
    IF(IREL.EQ.1)WRITE(8,659)
    RETURN
591  FORMAT(1H1,//////////)
    *5X, '*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 01) :',//
    *15X, ' 0 - INTERROMPER OS TRABALHOS',/
    *15X, ' 1 - GERAR UM GRAFO ALEATORIO',/
    *15X, ' 2 - CHAMAR GRAFO ARMAZENADO',/
    *15X, ' 3 - FORNECER GRAFO PELO TERMINAL',/
    *15X, ' 4 - ARMAZENAR GRAFO',/
    *15X, ' 5 - GRAFO PARTE FORNECIDO PARTE GERADO',/
    *15X, ' 6 - SUCESSORES DOS VERTICES',/
```

```
*15X,' 7 - SEMIGRAU EXTERIOR DE CADA VERTICE',/  
*15X,' 8 - IMPRIMIR MATRIZ DE ADJACENCIA',/  
*15X,' 9 - OBSERVAR MATRIZ DE ADJACENCIA, MAXIMO 15 VERTICE  
5',/  
*15X,'10 - PROXIMA LISTA DE OPCOES')  
592  FORMAT(1H1,//////////  
*5X,'*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 02) :',//  
*15X,'11 - VALORAR UM GRAFO',/  
*15X,'12 - CAMINHO MINIMO(FLOYD)',/  
*15X,'13 - DETERMINA A EXISTENCIA DE CIRCUITO(DEMOUCRON)',/  
*15X,'14 - CAMINHO MAXIMO(FLOYD)',/  
*15X,'15 - RAI0,DIAMETRO(GRAFOS NAO VALORADOS)',/  
*15X,'16 - COMPONENTES F-CONEXAS (MALGRANGE)',/  
*15X,'17 - RAI0,DIAMETRO (GRAFOS VALORADOS)',/  
*15X,'18 - FLUXO MAXIMO (FORD-FULKERSON)',/  
*15X,'19 - CAMINHO MINIMO (DIJKSTRA)',/  
*15X,'20 - PROXIMA LISTA DE OPCOES')  
593  FORMAT(1H1,//////////  
*5X,'*** ESCOLHA UMA DAS OPCOES ABAIXO (LISTA 03) :',//  
*15X,'21 - ARVORE PARCIAL MINIMA (KRUSKAL)',/  
*15X,'22 - TORNAR UM GRAFO, NAO ORIENTADO',/  
*15X,'23 - LIVRE',/  
*15X,'24 - LIVRE',/  
*15X,'25 - LIVRE',/  
*15X,'26 - LIVRE',/  
*15X,'27 - LIVRE',/  
*15X,'28 - LIVRE',/  
*15X,'29 - LIVRE',/  
*15X,'30 - VOLTA A LISTA 01(ESTA E A ULTIMA LISTA DE OPCOES  
)')  
599  FORMAT(//5X,'*** O GRAFO AINDA NAO FOI DEFINIDO',/9X,'ESCO  
LHA UMA  
*OPCAO QUE PERMITA A EDICAO DO GRAFO',/9X,'OPCOES 1,2,3 OU  
5')  
600  FORMAT(1H1,///5X,'*** PARABENS, VOCE NAO NECESSITOU NENHUM  
A REENTR  
*ADA')  
601  FORMAT(///20X,'*** FIM ***',///5X,'*** AGUARDE MENSAGEM DO  
SISTEMA  
*',//9X,'APOS ENTRE BYE')  
602  FORMAT(//5X,'*** DEVIDO A OPCAO = 0 OS TRABALHOS SERAO INT  
ERROMPID  
*OS')  
603  FORMAT(//5X,'*** PARA GRAVAR ENTRE COM A SENHA AUTORIZADA'  
)
```

```
604  FORMAT(/5X,'*** SENHA NAO AUTORIZADA,TENTE NOVAMENTE')
605  FORMAT(/5X,'*** DESISTA, ESCOLHA OUTRA OPCAO')
606  FORMAT(/5X,'*** ENTRE COM O NUMERO DE ARCOS/ARESTAS DO GR
    AFO',/9X
    *,'CASO NAO DESEJE LIMITAR ENTRE COM ZERO')
607  FORMAT(/5X,'*** PREFERE ESCOLHER OUTRA OPCAO ? SIM OU NAO
    ')
608  FORMAT(/5X,'*** ENTRE COM A SEMENTE ALEATORIA')
609  FORMAT(/5X,'*** ENTRE COM A PROBABILIDADE DA EXISTENCIA D
    E ARCOS/
    *ARESTAS')
610  FORMAT(/17X,'E ORIENTADO')
611  FORMAT(/17X,'NAO E ORIENTADO')
612  FORMAT(/5X,'*** PREFERE GERAR UM GRAFO DIFERENTE ? SIM OU
    NAO')
613  FORMAT(/5X,'*** PREFERE UMA VALORACAO DIFERENTE ? SIM OU
    NAO')
614  FORMAT(/5X,'*** DESEJA ENTRAR COM O VERTICE NOVAMENTE ? S
    IM OU NA
    *O')
615  FORMAT(/5X,'*** PREFERE LER OUTRO GRAFO ? SIM OU NAO')
616  FORMAT(1H1,15X,'*** SUCESSORES DOS VERTICES',///)
617  FORMAT(1H1,12X,'*** TOTAL DE SUCESSORES DOS VERTICES',///,
    4(5X,'VE
    *RT  NSUC'),//)
618  FORMAT(/5X,'*** FORAM IMPRESSOS OS RESULTADOS PEDIDOS',/9
    X,'O ESC
    *ANINHO TEM O NUMERO DO TERMINAL EM USO')
619  FORMAT(/5X,'*** DESEJA EXECUTAR OUTRA OPCAO ? SIM OU NAO'
    )
620  FORMAT(/5X,'*** ENTRADA ERRADA, TENTE NOVAMENTE')
621  FORMAT(/5X,'*** ENTRE COM O NUMERO DO GRAFO PADRONIZADO')
622  FORMAT(/5X,'*** PREFERE CHAMAR OUTRO GRAFO PADRONIZADO ?
    SIM OU N
    *AO')
623  FORMAT(/5X,'*** DESEJA OBSERVAR A MATRIZ DE ADJACENCIA ?
    SIM OU N
    *AO',/9X,'IMPORTANTE : SO VALIDO PARA GRAFOS ATE 15 VERTICE
    S')
624  FORMAT(/5X,'*** DESEJA SABER O SEMIGRAU EXTERIOR DOS V
    ERTICES
    *? SIM OU NAO')
625  FORMAT(/5X,'*** DESEJA SABER OS SUCESSORES DOS VERTICES ?
    SIM OU
    *NAO')
```

```
626  FORMAT(//5X,'*** O GRAFO E ORIENTADO ? SIM OU NAO')
627  FORMAT(//5X,'*** ENTRE COM O NUMERO DE VERTICES DO GRAFO,
      MAXIMO 4
      *0')
628  FORMAT(//5X,'*** FORNECA INICIALMENTE A PARTE DO GRAFO QUE
      VAI SER
      * LIDA',/9X,'MESMO OS VERTICES SEM SUCESSORES E/OU ANTECESS
      ORES',/9
      *X,'DEVEM SER INCLUIDOS')
629  FORMAT(//5X,'*** PREFERE MONTAR OUTRO GRAFO ? SIM OU NAO')
630  FORMAT(//5X,'*** DESEJA DAR UM VALOR CONSTANTE A TODOS ARC
      OS/AREST
      *AS ? SIM OU NAO')
631  FORMAT(//5X,'*** ENTRE COM O VALOR DA CONSTANTE')
632  FORMAT(//5X,'*** SEU GRAFO JA FOI VALORADO')
633  FORMAT(//5X,'*** DESEJA IMPRIMIR A MATRIZ DE VALORES ? SIM
      OU NAO')
      *)
634  FORMAT(//5X,'*** O SEU GRAFO AINDA NAO FOI VALORADO',/9X,'
      A ROTINA
      * DE VALORIZACAO SERA CHAMADA',/9X,'POREM SERA INCREMENTADO
      UM ERRO
      * AO CONTADOR')
635  FORMAT(//5X,'*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIA ? S
      IM OU NA
      *0')
636  FORMAT(//5X,'*** DESEJA IMPRIMIR A MATRIZ DE ROTEAMENTO
      ? SIM OU
      * NAO')
637  FORMAT(//5X,'*** INDIQUE ENTRE QUE VERTICES I=J DESEJA O C
      AMINHO M
      *INIMO(MAXIMO)')
638  FORMAT(//5X,'*** DESEJA OBTER O CAMINHO ENTRE OUTROS VERTI
      CES ? SI
      *M OU NAO')
639  FORMAT(//5X,'*** SEU GRAFO APRESENTA PELO MENOS UM CI
      RCUITO O
      *U CICLO')
640  FORMAT(//5X,'*** DESEJA SABER OS NIVEIS EXISTENTES NO GRAFO
      O ? SIM
      *OU NAO')
641  FORMAT(//5X,'*** NIVEIS EXISTENTES NO GRAFO',/)
642  FORMAT(//5X,'*** DEVIDO A EXISTENCIA DE CIRCUITO',/9X,'N
      AO PODE
      *SER CALCULADO O CAMINHO MAXIMO NO GRAFO')
```

```
643  FORMAT(//1X,'*** DESEJA IMPRIMIR A MATRIZ DE DISTANCIAS E
      AFASTAME
      *NTOS ? SIM OU NAO')
644  FORMAT(//5X,'*** OS NIVEIS EXISTENTES ATE O MOMENTO SAO : '
      ,//)
645  FORMAT(1H1,5X,'*** COMPONENTES F-CONEXAS DO GRAFO')
646  FORMAT(//5X,'*** O SEU GRAFO E FORTEMENTE CONEXO')
647  FORMAT(//5X,'*** DESEJA DAR VALORES AOS VERTICES ? SIM OU
      NAO')
648  FORMAT(//5X,'*** DESEJA IMPRIMIR OS VALORES DOS ARCOS',/9X
      , 'PARA O
      * FLUXO MAXIMO ? SIM OU NAO')
649  FORMAT(/5X,'*** AUSENCIA DE CENTROS EXTERIORES')
650  FORMAT(/5X,'*** AUSENCIA DE CENTROS INTERIORES')
651  FORMAT(//5X,'*** ENTRE COM O VALOR DA RAIZ DESEJADA')
652  FORMAT(//5X,'*** DESEJA MUDAR O VALOR DA RAIZ ? SIM OU NAO
      ')
653  FORMAT(/5X,'*** NAO EXISTE NENHUM GRAFO NO ARQUIVO')
654  FORMAT(/5X,'*** O GRAFO NAO APRESENTA NENHUM CICLO')
655  FORMAT(//5X,'*** ALGORITMO SO E VALIDO PARA GRAFOS NAO ORI
      ENTADOS'
      *,/9X,'DESEJA DESORIENTAR O GRAFO ? SIM OU NAO')
656  FORMAT(//5X,'*** DESEJA SABER OS ARCOS DA ARVORE PARCIAL M
      INIMA ?
      *SIM OU NAO')
657  FORMAT(1H1, //5X, '*** ARCOS DA ARVORE PARCIAL MINIMA (KRUSK
      AL)', /)
658  FORMAT(//5X, '*** PARA SABER OS VERTICES DO CAMINHO USAR FL
      OYD (OPC
      *AO=12)')
659  FORMAT(//5X, '*** O GRAFO JA E NAO ORIENTADO')
      END
```

```
C
C  MONTAGEM DOS FORMATOS PARA LEITURA
```

```
C
      SUBROUTINE LINHA(N)
      DIMENSION AC(3),FC(3)
      DATA AC,FC,FORMS,ETX/3*Z1F,3*Z1E,Z12,Z03/
      GO TO (100,200,300,400,500,600),N
100  WRITE(6,10)(AC(I),FC(I),I=1,3),FORMS,ETX
10   FORMAT(1X,C1,40X,2C1,3X,C1,5X,'XMT',C1,1X,3C1)
      RETURN
200  WRITE(6,20)(AC(I),FC(I),I=1,2),FORMS,ETX
20   FORMAT(1X,C1,2X,C1,5X,'XMT',C1,1X,3C1)
      RETURN
```

```
300 WRITE(6,30)(AC(I),FC(I),I=1,2),FORMS,ETX
30  FORMAT(1X,C1,3X,C1,5X,'XMT',C1,1X,3C1)
   RETURN
400 WRITE(6,40)(AC(I),FC(I),I=1,2),FORMS,ETX
40  FORMAT(1X,C1,4X,C1,5X,'XMT',C1,1X,3C1)
   RETURN
500 WRITE(6,50)(AC(I),FC(I),I=1,2),FORMS,ETX
50  FORMAT(1X,C1,6X ,C1,5X,'XMT',C1,1X,3C1)
   RETURN
600 WRITE(6,60)(AC(I),FC(I),I=1,3),FORMS,ETX
60  FORMAT(1X,C1,2X,2C1,2X,C1,5X,'XMT',C1,1X,3C1)
   RETURN
END
```

```
C
C  MONTAGEM DOS FORMATOS PARA LEITURA COM K POSSIBILIDADES
C
```

```
   SUBROUTINE LER(N,K)
   DIMENSION AC(10),FC(10)
   DATA AC,FC,FORMS,ETX/10*Z1F,10*Z1E,Z12,Z03/
   GO TO (100,200,400),N
100  WRITE(6,10)(AC(I),FC(I),I=1,K)
10   FORMAT(10(1X,C1,2X,C1))
   GO TO 300
200  WRITE(6,20)(AC(I),FC(I),I=1,K)
20   FORMAT(1X,9(C1,6X,C1))
   GO TO 300
 400  WRITE(6,40)(AC(I),FC(I),I=1,K)
 40   FORMAT(1X,10(C1,5X,C1))
300  WRITE(6,30)AC(1),FC(1),FORMS,ETX
30   FORMAT(1X,'XMT',C1,1X,3C1)
   RETURN
END
```

```
C
C  PERMITE FAZER A PARTICAO EM NIVEIS (DEMOUCRON)
C
```

```
   SUBROUTINE DMCRON
   DIMENSION MADJ(40,40),NSUC(40)
   COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
   COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
   DIMENSION NIVEL(40,40),KTEST(40),KEY(40),MAX(40)
25  FORMAT(1H1,////////////////////////////////////)
   ICIRC=0
   DO 100 I=1,NVER
100  KEY(I)=0
   N=0
```



```
303 K1=0
    DO 200 J=1,NVER
    KTEST(J)=0
    IF(KEY(J))200,201,200
201 DO 200 I=1,NVER
    IF(KEY(I))200,202,200
202 KTEST(J)=KTEST(J)+MADJ(I,J)
    K1=1
200 CONTINUE
    IF(IOR.EQ.1)GO TO 305
    DO 400 I=1,NVER
    IF(KTEST(I).EQ.1)KTEST(I)=0
    400 CONTINUE
305 IF(K1)203,204,203
203 N=N+1
    IV=0
    K2=0
    DO 300 J=1,NVER
    IF(KEY(J).NE.0.OR.KTEST(J).NE.0)GO TO 300
    IV=IV+1
    MAX(N)=IV
    NIVEL(N,IV)=J
    KEY(J)=1
    K2=1
300 CONTINUE
    IF(K2)303,304,303
    304 ICIRC=1
    IF(IOPC.EQ.21)RETURN
    IF(N=1)306,307,306
    307 CALL MSG(639,IREL)
    GO TO 251
    306 N=N-1
    CALL MSG(639,IREL)
    CALL MSG(644,IREL)
    I=1
    GO TO 205
    204 IF(IOPC.EQ.21)RETURN
    IF(IOR.EQ.1)GO TO 207
    CALL MSG(654,IREL)
    GO TO 251
    207 WRITE(6,25)
    I=1
    CALL MSG(640,IREL)
    CALL SIND(K1,&250,&252,&204)
250 CALL MSG(641,IREL)
```

```
205 DO 206 K=1,10
      WRITE(6,30)I,(NIVEL(I,J),J=1,MAX(I))
      IF(IREL.EQ.1)WRITE(8,30)I,(NIVEL(I,J),J=1,MAX(I))
30   FORMAT(/2X,'NIVEL',I3,' ...',4(T17,10I4,/))
      I=I+1
      IF(I.GT.N)GO TO 251
206  CONTINUE
      CALL PAUSA
      GO TO 250
      251 CALL PAUSA
252  IF(IOPC.EQ.13)CALL NEWOP
      RETURN
      END
```

```
C
C   PERMITE ACHAR O CAMINHO MAXIMO ENTRE DOIS VERTICES(FLOYD)
C
```

```
      SUBROUTINE FLOYD2
      DIMENSION MADJ(40,40),NSUC(40)
      DIMENSION VAL(40,40),DIST(40,40),MROT(40,40)
      DIMENSION LCAM(40)
      COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
      COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
      COMMON /BLK4/IVLR,VAL
25   FORMAT(1H1,////////////////////////////////////)
      IF(IVLR.EQ.1)GO TO 100
      CALL MSG(634,IREL)
      CALL PAUSA
      NERRO=NERRO+1
      CALL VALORA
100  CALL DMCRON
      IF(ICIRC.EQ.0)GO TO 101
      CALL MSG(642,IREL)
      CALL PAUSA
      CALL NEWOP
      RETURN
101  DO 150 I=1,NVER
      DO 150 J=1,NVER
      IF(VAL(I,J)+1E67)140,151,151
140  MROT(I,J)=J
      DIST(I,J)=VAL(I,J)
      GO TO 150
151  MROT(I,J)=0
      DIST(I,J)=(-1.)*(1E68)
150  CONTINUE
      DO 200 K=1,NVER
```

```
DO 200 I=1,NVER
DO 200 J=1,NVER
IF(DIST(I,J)-(DIST(I,K)+DIST(K,J)))201,200,200
201 DIST(I,J)=DIST(I,K)+DIST(K,J)
MROT(I,J)=MROT(I,K)
200 CONTINUE
450 WRITE(6,25)
CALL MSG(635,IREL)
CALL SINO(K2,8250,8350,8450)
250 WRITE(8,12)NVER,NVER,(I,I=1,NVER)
12 FORMAT(1H1, //40X, 'MATRIZ DE DISTANCIA MAXIMA DO GRAFO', I3,
' X', I3,
*//,4(10X,10(12,9X),/))
DO 300 I=1,NVER
WRITE(8,13)I,(DIST(I,J),J=1,NVER)
13 FORMAT(/14,1X,4(10(F9.2,2X),/5X))
300 CONTINUE
CALL MSG(618,IREL)
CALL PAUSA
350 WRITE(6,25)
CALL MSG(636,IREL)
CALL SINO(K2,8550,8650,8350)
550 WRITE(8,15)NVER,NVER,(I,I=1,NVER)
15 FORMAT(1H1, //30X, 'MATRIZ DE ROTEAMENTO MAXIMO DO GRAFO', I3
, ' X', I3
*,//10X,40I3)
DO 400 I=1,NVER
WRITE(8,16)I,(MROT(I,J),J=1,NVER)
16 FORMAT(/5X, I3, 2X, 40I3)
400 CONTINUE
CALL MSG(618,IREL)
CALL PAUSA
650 WRITE(6,25)
CALL MSG(637,IREL)
CALL LINHA(6)
READ(5,10,DATA=99)I,J
10 FORMAT(2I2)
CALL NEG(I,8050)
CALL NEG(J,8050)
IF(I.LE.NVER.AND.J.LE.NVER)GO TO 750
WRITE(6,17)I,J
IF(IREL.EQ.1)WRITE(8,17)I,J
17 FORMAT(/5X, '*** VALORES LIDOS I=', I3, 2X, 'J=', I3, 1X, 'INCOM
PATIVEL
*COM NUMERO DE VERTICES', /9X, 'TENDE NOVAMENTE')
```

```

    NERRO=NERRO+1
    CALL PAUSA
    GO TO 650
99  CALL MSG(620,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    GO TO 650
750 IF(DIST(I,J).LE.-1E67)GO TO 950
    K=1
    LCAM(K)=I
851 KK=LCAM(K)
    IF(MROT(KK,J).EQ.J)GO TO 850
    K=K+1
    LCAM(K)=MROT(KK,J)
    GO TO 851
850 K=K+1
    LCAM(K)=J
    WRITE(6,18)I,J,(LCAM(M),M=1,K)
    IF(IREL.EQ.1)WRITE(8,18)I,J,(LCAM(M),M=1,K)
18  FORMAT(/5X,'*** VARTICES DO CAMINHO MAXIMO ENTRE',2X,'( ',
    I2,' E ',
    *I3,' )',4(/9X,10T4))
    WRITE(6,19)I,J,DIST(I,J)
    IF(IREL.EQ.1)WRITE(8,19)I,J,DIST(I,J)
19  FORMAT(/5X,'*** DISTANCIA MAXIMA ENTRE OS VERTICES',2X,'( ',
    I2,' E ',
    *I3,' )',2X,F9.2)
    CALL PAUSA
    GO TO 951
950 WRITE(6,20)I,J
    IF(IREL.EQ.1)WRITE(8,20)I,J
20  FORMAT(/5X,'*** NAO EXISTE NENHUM CAMINHO ENTRE OS VERTIC
    ES',2X,'
    *(',I2,' E ',I3,' )')
    CALL PAUSA
951 WRITE(6,25)
    CALL MSG(630,IREL)
    CALL SINO(K2,&650,&952,&951)
952 IF(TOPC.EQ.14)CALL NEWOP
    RETURN
    END
```

```

C
C   RAIO,DIAMETRO,AFASTAMENTO,CENTROS E PERIFERICOS
C   PARA GRAFOS NAO VALORADOS
C
```

```
SUBROUTINE RADI
DIMENSION MADJ(40,40),NSUC(40)
DIMENSION MDIST(40,40),IAFAST(40),IPER(40),ICENT(40)
DIMENSION JAFAST(40),JPER(40),JCENT(40),LABS(40),LBID(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
25  FORMAT(1H1,////////////////////)
DO 100 I=1,NVER
DO 100 J=1,NVER
MDIST(I,J)=MADJ(I,J)
IF(MADJ(I,J).EQ.0)MDIST(I,J)=10000
100  CONTINUE
DO 200 K=1,NVER
DO 200 I=1,NVER
DO 200 J=1,NVER
IF(MDIST(I,J)-(MDIST(I,K)+MDIST(K,J)))200,200,201
201  MDIST(I,J)=MDIST(I,K)+MDIST(K,J)
200  CONTINUE
DO 300 I=1,NVER
MDIST(I,I)=0
DO 300 J=1,NVER
JAFAST(I)=MAX0(JAFAST(I),MDIST(J,I))
LBID(I)=MAX0(LBID(I),MDIST(I,J)+MDIST(J,I))
300  IAFAST(I)=MAX0(IAFAST(I),MDIST(I,J))
IRAIO=10000
IDIAM=0
JRAIO=10000
JDIAM=0
IRID=10000
DO 400 I=1,NVER
JDIAM=MAX0(JDIAM,JAFAST(I))
JRAIO=MIN0(JRAIO,JAFAST(I))
IRID=MIN0(IRID,LBID(I))
IDIAM=MAX0(IDIAM,IAFAST(I))
400  IRAIO=MIN0(IRAIO,IAFAST(I))
IMC=0
IMP=0
JMC=0
JMP=0
IBI=0
DO 500 I=1,NVER
IF(IAFAST(I).NE.IPAIO)GO TO 501
IMC=IMC+1
ICENT(IMC)=I
501  IF(IAFAST(I).NE.IDIAM)GO TO 502
```

```
IMP=IMP+1
IPER(IMP)=I
502 IF(JAFAS(T).NE.JPAIO)GO TO 503
JMC=JMC+1
JCENT(JMC)=I
503 IF(JAFAS(T).NE.JDIAM)GO TO 504
JMP=JMP+1
JPER(JMP)=I
504 IF(LBID(I).NE.IBID)GO TO 500
IBI=IBI+1
LABS(IBI)=I
500 CONTINUE
350 WRITE(6,25)
CALL MSG(643,IREL)
CALL SINO(K2,8150,8250,8350)
150 WRITE(8,10)NVER,NVER,(K,K=1,NVER)
10 FORMAT(1H1, //35X, 'MATRIZ DE DISTANCIAS DO GRAFO ',I3, ' X',
I3, ///10
*X,40I3)
DO 600 I=1,NVER
600 WRITE(8,20)I,(MDIST(I,J),J=1,NVER)
20 FORMAT(/5X,I3,2X,40(1X,I2))
WRITE(8,30) (I,IAFAST(I),I=1,NVER)
30 FORMAT(1H1,30X,'AFASTAMENTOS EXTERIORES DOS VERTICES',//4(
5X,'VERT
*ICE AFASTAMENTO'),//10(4(8X,I2,10X,I2,3X),//))
WRITE(8,31)(I,JAFAST(I),I=1,NVER)
31 FORMAT(///30X,'AFASTAMENTOS INTERIORES DOS VERTICES',//4(5
X,'VERTI
*CE AFASTAMENTO'),//10(4(8X,I2,10X,I2,3X),//))
CALL MSG(618,IREL)
CALL PAUSA
250 WRITE(6,40)IRAIO,IDIAM
IF(IREL.EQ.1)WRITE(8,40)IRAIO,IDIAM
40 FORMAT(1H1,5X,'*** INFORMACOES SOBRE O GRAFO',//5X,'RAIO E
XTERIOR'
*,2X,I4, //5X, 'DIAMETRO EXTERIOR',2X,I4)
IF(IRAIO.NE.10000)GO TO 450
CALL MSG(640,IREL)
GO TO 550
450 WRITE(6,60)(ICENT(I),I=1,IMC)
IF(IREL.EQ.1)WRITE(8,60)(ICENT(I),I=1,IMC)
60 FORMAT(/5X,'CENTROS EXTERIORES',4(T25,10I4,/)
WRITE(6,70)(IPER(I),I=1,IMP)
IF(IREL.EQ.1)WRITE(8,70)(IPER(I),I=1,IMP)
```

```
70  FORMAT(/5X,'PERIFERICOS EXTERIORES',4(T29,10I4,/))
550  WRITE(6,41)JRAIO,JDIAM
    IF(IREL.EQ.1)WRITE(8,41)JRAIO,JDIAM
41   FORMAT(//5X,'RAIO INTERIOR',2X,I4,//5X,'DIAMETRO INTERIOR
    ',2X,I4)
    IF(JRAIO.NE.10000)GO TO 451
    CALL MSG(659,IREL)
    GO TO 551
451  WRITE(6,61)(JCENT(I),I=1,JMC)
    IF(IREL.EQ.1)WRITE(8,61)(JCENT(I),I=1,JMC)
61   FORMAT(/5X,'CENTROS INTERIORES',4(T25,10I4,/))
    WRITE(6,71)(JPER(I),I=1,JMP)
    IF(IREL.EQ.1)WRITE(8,71)(JPER(I),I=1,JMP)
71   FORMAT(/5X,'PERIFERICOS INTERIORES',4(T29,10I4,/))
    IF(IBID.EQ.10000)GO TO 551
    WRITE(6,81)(LABS(I),I=1,IBI)
    IF(IREL.EQ.1)WRITE(8,81)(LABS(I),I=1,IBI)
81   FORMAT(/5X,'CENTROS BI-DIRECIONAIS',4(T29,10I4,/))
551  CALL PAUSA
    IF(IOPC.EQ.15)CALL NEWOP
    RETURN
    END
```

```
C
C   RAIO,DIAMETRO,AFASTAMENTOS,CENTROS E PERIFERICOS
C   PARA GRAFOS VALORADOS
C
```

```
      SUBROUTINE RADIV
      DIMENSION MADJ(40,40),NSUC(40),VAL(40,40),PESO(40),DIST(40
,40)
      DIMENSION AFASTI(40),AFASTJ(40),ICENT(40),JCENT(40)
      DIMENSION IPER(40),JPER(40),BID(40),LABS(40)
      COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
      COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
      COMMON /BLK4/IVLR,VAL
25   FORMAT(1H1,////////////////////////////////////)
    IF(IVLR.EQ.1)GO TO 100
    CALL MSG(634,IREL)
    CALL PAUSA
    NERRO=NERRO+1
    CALL VALORA
100  WRITE(6,25)
    CALL MSG(647,IREL)
    CALL SINO(K1,8101,8102,8100)
102  DO 180 I=1,NVER
180  PESO(I)=1.0
```

```
GO TO 200
101 N=NVER/10
    N1=N*10
    IF(N1.NE.NVER)N=N+1
    DO 121 K=1,N
    I2=K*10
    IF(K.EQ.N)I2=NVER
    I1=K*10-9
201 WRITE(6,25)
    WRITE(6,26)I1,I2
26  FORMAT(/5X,'*** ENTRE COM OS VALORES DO VERTICE',I3,1X,'A
    TE O VER
    *TICE',I3)
    N2=I2-I1+1
    CALL LER(3,N2)
    READ(5,10,DATA=99)(PESO(I),I=I1,I2)
10  FORMAT(10F5.2)
    GO TO 121
99  CALL MSG(620,IREL)
    NERRO=NERRO+1
    CALL PAUSA
    GO TO 201
121 CONTINUE
    WRITE(6,22)(I,PESO(I),I=1,NVER)
    IF(IRL.EQ.1)WRITE(8,22)(I,PESO(I),I=1,NVER)
22  FORMAT(1H1,20X,'VALORES LIDOS PARA OS VERTICES',//4(2X,'VE
    RT
    *VALOR'),//10(4(3X,I2,F12.2),//))
    CALL PAUSA
200 DO 160 I=1,NVER
    DO 160 J=1,NVER
160 DIST(I,J)=VAL(I,J)
    DO 170 K=1,NVER
    DO 170 I=1,NVER
    DO 170 J=1,NVER
    IF(DIST(I,K).GE.1E67.OR.DIST(K,J).GE.1E67)GO TO 170
    IF(DIST(I,J)=(DIST(I,K)+DIST(K,J)))170,170,171
171 DIST(I,J)=DIST(I,K)+DIST(K,J)
170 CONTINUE
    DO 300 I=1,NVER
    DIST(I,I)=0.
    DO 300 J=1,NVER
    IF(DIST(I,J).GE.1F67)GO TO 302
    AFASTI(I)=AMAXI(AFASTI(I),PESO(J)*DIST(I,J))
    GO TO 303
```



```
302  AFASTI(I)=1E68
303  IF(DIST(J,I).GE.1E67)GO TO 305
    AFASTJ(I)=AMAX1(AFASTJ(I),PESO(J)*DIST(J,I))
    GO TO 306
305  AFASTJ(I)=1E68
306  IF(DIST(I,J).GE.1E67.OR.DIST(J,I).GE.1E67)GO TO 307
    BID(I)=AMAX1(BID(I),PESO(J)*(DIST(I,J)+DIST(J,I)))
    GO TO 300
307  BID(I)=1E68
300  CONTINUE
    RAI0I=1E68
    DIAMI=0.
    RAI0J=1E68
    DIAMJ=0.
    BIDI=1E68
    DO 400 I=1,NVER
    DIAMJ=AMAX1(DIAMJ,AFASTJ(I))
    RAI0J=AMIN1(RAI0J,AFASTJ(I))
    BIDI=AMIN1(BIDI,BID(I))
    DIAMI=AMAX1(DIAMI,AFASTI(I))
400  RAI0I=AMIN1(RAI0I,AFASTI(I))
    IMC=0
    IMP=0
    JMC=0
    JMP=0
    IBI=0
    DO 500 I=1,NVER
    IF(AFASTI(I).NE.RAI0I)GO TO 501
    IMC=IMC+1
    ICENT(IMC)=I
501  IF(AFASTI(I).NE.DIAMI)GO TO 502
    IMP=IMP+1
    IPER(IMP)=I
502  IF(AFASTJ(I).NE.RAI0J)GO TO 503
    JMC=JMC+1
    JCENT(JMC)=I
503  IF(AFASTJ(I).NE.DIAMJ)GO TO 504
    JMP=JMP+1
    JPER(JMP)=I
504  IF(BID(I).NE.BIDI)GO TO 500
    IBI=IBI+1
    LABS(IBI)=I
500  CONTINUE
350  WRITE(6,25)
    CALL MSG(643,IREL)
```

```
CALL SINO(KI, &150, &250, &350)
150 WRITE(8, 15) NVER, NVER, (K, K=1, NVER)
15  FORMAT(1H1, //40X, 'MATRIZ DE DISTANCIAS DO GRAFO', I3, ' X', I
3, //4(10
*X, 10(I2, 9X), //))
DO 600 I=1, NVER
600 WRITE(8, 20) I, (DIST(I, J), J=1, NVER)
20  FORMAT(/I4, 1X, 4(10(F9.2, 2X), /5X))
WRITE(8, 30) (I, AFASTI(I), I=1, NVER)
30  FORMAT(1H1, 30X, 'AFASTAMENTOS EXTERIORES DOS VERTICES', //4(
5X, 'VERT
*ICE AFASTAMENTO'), //10(4(8X, I2, 6X, F9.2), //))
WRITE(8, 31) (I, AFASTJ(I), I=1, NVER)
31  FORMAT(///30X, 'AFASTAMENTOS INTERIORES DOS VERTICES', //4(5
X, 'VERTI
*CE AFASTAMENTO'), //10(4(8X, I2, 6X, F9.2), //))
CALL MSG(618, IREL)
CALL PAUSA
250 WRITE(6, 40) RAI0I, DIAMI
IF (IREL.EQ.1) WRITE(8, 40) RAI0I, DIAMI
40  FORMAT(1H1, 5X, '*** INFORMACOES SOBRE O GRAFO', //5X, 'RAIO E
XTERIOR'
*, 2X, F9.2, //5X, 'DIAMETRO EXTERIOR', 2X, F9.2)
IF (RAI0I.LE.1E67) GO TO 450
CALL MSG(649, IREL)
GO TO 550
450 WRITE(6, 60) (ICENT(I), I=1, IMC)
IF (IREL.EQ.1) WRITE(8, 60) (ICENT(I), I=1, IMC)
60  FORMAT(/5X, 'CENTROS EXTERIORES', 4(T25, 1@I4, /))
WRITE(6, 70) (IPER(I), I=1, IMP)
IF (IREL.EQ.1) WRITE(8, 70) (IPER(I), I=1, IMP)
70  FORMAT(/5X, 'PERIFERICOS EXTERIORES', 4(T29, 1@I4, /))
550 WRITE(6, 41) RAI0J, DIAMJ
IF (IREL.EQ.1) WRITE(8, 41) RAI0J, DIAMJ
41  FORMAT(///5X, 'RAIO INTERIOR', 2X, F9.2, //5X, 'DIAMETRO INTERI
OR', 2X,
*f9.2)
IF (RAI0J.LE.1E67) GO TO 451
CALL MSG(650, IREL)
GO TO 551
451 WRITE(6, 61) (JCENT(I), I=1, JMC)
IF (IREL.EQ.1) WRITE(8, 61) (JCENT(I), I=1, JMC)
61  FORMAT(/5X, 'CENTROS INTERIORES', 4(T25, 1@I4, /))
WRITE(6, 71) (JPER(I), I=1, JMP)
IF (IREL.EQ.1) WRITE(8, 71) (JPER(I), I=1, JMP)
```

```
71  FORMAT(/5X,'PERIFERICOS INTERIORES',4(T29,10I4,/))
    IF(BIDI.GT.1E67)GO TO 551
    WRITE(6,81)(LABS(I),I=1,IBI)
    IF(IREL.EQ.1)WRITE(8,81)(LABS(I),I=1,IBI)
81  FORMAT(/5X,'CENTROS BIDIRECIONAIS',4(T29,10I4,/))
551  CALL PAUSA
    IF(IOPC.EQ.17)CALL NEWOP
    RETURN
    END
```

```
C
C  DECOMPOSICAO EM COMPONENTES F-CONEXAS (MALGRANGE)
C
```

```
    SUBROUTINE FCONEX
    DIMENSION MADJ(40,40),NSUC(40),KEY(40)
    INTEGER GRAU(40),GMAIS(40),GMENOS(40),COMP(40)
    COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
    COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
    K=1
    NORD=1
550  KK=K
    DO 100 I=1,NVER
100  GRAU(I)=NSUC(I)
    GMAIS(K)=1
150  DO 200 J=1,NVER
    IF(MADJ(K,J).EQ.0)GO TO 200
    KEY(J)=J
    GMAIS(J)=1
    GRAU(K)=GRAU(K)-1
200  CONTINUE
    DO 300 J=1,NVER
    IF(KEY(J).EQ.0)GO TO 300
    IF(GRAU(J).LE.0)GO TO 350
    K=J
    KEY(J)=0
    GO TO 150
350  KEY(J)=0
300  CONTINUE
    DO 400 I=1,NVER
    DO 400 J=1,NVER
400  GRAU(I)=GRAU(I)+MADJ(J,I)
    K=KK
    GMENOS(K)=1
250  DO 500 J=1,NVER
    IF(MADJ(J,K).EQ.0)GO TO 500
    KEY(J)=J
```

```
GMENOS(J)=1
GRAU(K)=GRAU(K)-1
500 CONTINUE
DO 600 J=1,NVER
IF(KEY(J).EQ.0)GO TO 600
IF(GRAU(J).LE.0)GO TO 650
K=J
KEY(J)=0
GO TO 250
650 KEY(J)=0
600 CONTINUE
DO 700 I=1,NVER
GMAIS(I)=GMAIS(I)+GMENOS(I)
IF(GMAIS(I).EQ.1)GMAIS(I)=0
700 GMENOS(I)=GMAIS(I)
DO 800 I=1,NVER
IF(GMAIS(I).NE.0)GO TO 800
K=I
NORD=NORD+1
GO TO 550
800 CONTINUE
IF(NORD.EQ.1)GO TO 805
MIN=100
DO 850 I=1,NVER
850 MIN=MINO(MIN,GMAIS(I))
CALL MSG(645,IREL)
750 NORD=1
MINI=100
DO 900 I=1,NVER
IF(GMAIS(I).NE.MIN)GO TO 950
COMP(NORD)=I
NORD=NORD+1
N=N+1
GMAIS(I)=1000
GO TO 900
950 MINI=MINO(MINI,GMAIS(I)-MIN)
900 CONTINUE
WRITE(6,10)(COMP(J),J=1,NORD-1)
IF(IREL.EQ.1)WRITE(8,10)(COMP(J),J=1,NORD-1)
10 FORMAT(/2X,'VERTICES ...',4(T17,1014,/))
IF(N.EQ.NVER)GO TO 806
MIN=MIN+MINI
GO TO 750
805 CALL MSG(646,IREL)
806 CALL PAUSA
```

```
IF(IOPC.EQ.16)CALL NEWOP
RETURN
END

C
C FLUXO MAXIMO (FORD-FULKERSON)
C
SUBROUTINE FLUXO
DIMENSION MADJ(40,40),NSUC(40),VAL(40,40),X(40,40)
DIMENSION KEY(40,2),SINAL(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK4/IVLR,VAL
24  FORMAT(1H1,////////////////////)
IF(IVLR.EQ.1)GO TO 101
CALL MSG(634,IREL)
CALL PAUSA
NERRO=NERRO+1
CALL VALORA
101 DO 100 I=1,NVER
DO 100 J=1,NVER
X(I,J)=VAL(I,J)
IF(MADJ(I,J).EQ.0)X(I,J)=0.0
100 CONTINUE
50 DO 200 I=1,NVER
SINAL(I)=0.0
KEY(I,1)=0
200 KEY(I,2)=0
IW=0
SINAL(1)=1E68
I=1
850 J=2
550 IF(X(I,J).LE.0.0)GO TO 150
IF(SINAL(J).NE.0.0)GO TO 250
SINAL(J)=AMINI(SINAL(I),X(I,J))
KEY(J,1)=1
IF(J.LT.I)IW=1
IF(J=NVER)250,350,250
150 IF(J.EQ.NVER)GO TO 450
250 J=J+1
GO TO 550
450 KEY(I,2)=1
950 I=I+1
IF(I.NE.NVER)GO TO 650
IF(IW.EQ.0)GO TO 750
I=2
```

```

      IW=0
650  IF(SINAL(I).GT.0.0.AND.KEY(I,2).EQ.0)GO TO 850
      GO TO 950
350  X(I,J)=X(I,J)-SINAL(NVER)
      X(J,I)=X(J,I)+SINAL(NVER)
      J=I
      IF(J.EQ.1)GO TO 50
      I=KEY(J,1)
      GO TO 350
750  F=0.0
      DO 300 I=2,NVER
300  F=F+X(I,1)
      DO 400 I=1,NVER
      DO 400 J=1,NVER
      IF(MADJ(I,J).EQ.0)GO TO 400
      X(I,J)=VAL(I,J)-X(I,J)
400  CONTINUE
      WRITE(6,10)F
      IF(IREL.EQ.1)WRITE(8,10)F
10   FORMAT(1H1, //5X, '*** O FLUXO MAXIMO OBTIDO POR FORD-FULKER
      SON FOI'
      *, //3X, F10.2)
      CALL PAUSA
140  WRITE(6,20)
      CALL MSG(608,IREL)
      CALL SING(KK,8120,8130,8140)
120  WRITE(8,20)
20   FORMAT(1H1, //10X, 'VALORES DOS ARCOS PARA O FLUXO MAXIMO(FO
      RD-FULKE
      *RSON)', //)
      DO 500 I=1,NVER
      DO 500 J=1,NVER
      IF(X(I,J).LE.0.0.OR.MADJ(I,J).EQ.0)GO TO 500
      IF(X(I,J).EQ.VAL(I,J))GO TO 501
      WRITE(8,30)I,J,X(I,J)
30   FORMAT(/3X, 'ARCO',2I3, ' =', F10.2)
      GO TO 500
501  WRITE(8,40)I,J,X(I,J)
40   FORMAT(/3X, 'ARCO',2I3, ' =', F10.2,2X, 'SATURADO')
500  CONTINUE
      CALL MSG(618,IREL)
      CALL PAUSA
130  IF(IOPC.EQ.18)CALL NEWOP
      RETURN
      END
```

```
C
C CAMINHO MINIMO (DIJKSTRA)
C
SUBROUTINE DKSTRA
DIMENSION MADJ(40,40),NSUC(40)
DIMENSION VAL(40,40),DIST(40,40),MARK(40)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK4/IVLR,VAL
25  FORMAT(1H1,////////////////////)
IF(IVLR.EQ.1)GO TO 151
CALL MSG(634,IREL)
CALL PAUSA
NERRO=NERRO+1
CALL VALORA
151  WRITE(6,25)
CALL MSG(651,IREL)
CALL LINHA(2)
READ(5,10,DATA=99)NRAIZ
10  FORMAT(I2)
CALL NEG(NRAIZ,&151)
IF(NRAIZ.LE.NVER)GO TO 100
WRITE(6,11)NRAIZ
IF(IREL.EQ.1)WRITE(8,11)NRAIZ
11  FORMAT(/5X,'*** VALOR LIDO =',I5,2X,'MAIOR QUE NUMERO DE
VERTICES
*, TENTE NOVAMENTE')
NERRO=NERRO+1
CALL PAUSA
GO TO 151
99  CALL MSG(620,IREL)
CALL PAUSA
NERRO=NERRO+1
GO TO 151
100 DO 150 I=1,NVER
MARK(I)=0
DO 150 J=1,NVER
150  DIST(I,J)=VAL(I,J)
J1=NRAIZ
MARK(NRAIZ)=1
DO 200 K=1,NVER-1
CMIN=1E68
DO 300 J=1,NVER
IF(MARK(J).EQ.1.OR.DIST(NRAIZ,J1).GE.1E67.OR.DIST(J1,J).GE
.1E67)GO
```

```

* TO 300
DIST(NRAIZ,J)=AMIN1(DIST(NRAIZ,J),DIST(NRAIZ,J1)+DIST(J1,J
))
300 CONTINUE
DO 400 J=1,NVER
IF(MARK(J).EQ.1.OR.CMIN.LE.DIST(NRAIZ,J))GO TO 400
JJ=J
CMIN=DIST(NRAIZ,J)
400 CONTINUE
MARK(JJ)=1
J1=JJ
200 CONTINUE
I=0
156 WRITE(6,12)NRAIZ
IF(IREL.EQ.1)WRITE(8,12)NRAIZ
12 FORMAT(1H1, //5X, '*** DISTANCIAS ENTRE A RAIZ', I3, ' E OS V
ERTICES
*!', /)
154 I=I+1
WRITE(6,13)I,DIST(NRAIZ,I)
IF(IREL.EQ.1)WRITE(8,13)I,DIST(NRAIZ,I)
13 FORMAT(10X, 'VERTICE', I3, ' ... ', F10.2)
IF(I.EQ.NVER)GO TO 155
IF(I.EQ.15.OR.I.EQ.30)GO TO 157
GO TO 154
157 CALL PAUSA
GO TO 156
155 CALL MSG(658,IREL)
CALL PAUSA
153 WRITE(6,25)
CALL MSG(652,IREL)
CALL SINO(KK,8151,8152,8153)
152 IF(IOPC.EQ.19)CALL NEWOP
RETURN
END

C
C ARVORE PARCIAL DE VALOR MINIMO (KRUSKAL)
C
SUBROUTINE KRUSKA
DIMENSION MADJ(40,40),NSUC(40),VAL(40,40)
DIMENSION MT(40,40),IMIN(1560),JMIN(1560),VMIN(1560)
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK4/IVLR,VAL
24 FORMAT(1H1, ///////////////////////////////////////////////////)

```



```
25  FORMAT(1H1,////////////////////////////////////)
    IF(IOR.EQ.0)GO TO 100
    NFRRO=NERR0+1
103  WRITE(6,24)
    CALL MSG(655,IREL)
    CALL SINO(K1,8101,8102,8103)
101  CALL DISOR
100  IF(IVLR.EQ.1)GO TO 104
    CALL MSG(634,IREL)
    CALL PAUSA
    NERRD=NERR0+1
    CALL VALORA
104  DO 105 I=1,NVER
    DO 105 J=1,NVER
    MT(I,J)=MADJ(I,J)
    MADJ(I,J)=0
105  CONTINUE
    K=0
    DO 200 I=2,NVER
    DO 200 J=1,I-1
    IF(VAL(I,J).GE.1E67)GO TO 200
    K=K+1
    IMIN(K)=I
    JMIN(K)=J
    VMIN(K)=VAL(I,J)
200  CONTINUE
301  KEY=0
    DO 300 K=1,NARC-1
    IF(VMIN(K).LE.VMIN(K+1))GO TO 300
    IT=IMIN(K)
    JT=JMIN(K)
    VT=VMIN(K)
    VMIN(K)=VMIN(K+1)
    IMIN(K)=IMIN(K+1)
    JMIN(K)=JMIN(K+1)
    IMIN(K+1)=IT
    JMIN(K+1)=JT
    VMIN(K+1)=VT
    KEY=1
300  CONTINUE
    IF(KEY.EQ.1)GO TO 301
    KFIM=NVER-1
    IF(KFIM.GT.NARC)KFIM=NARC
    K=1
    DO 500 KK=1,KFIM
```

```
I=IMIN(K)
J=JMIN(K)
K=K+1
MADJ(I,J)=1
MADJ(J,I)=1
CALL DMCRON
IF(ICIRC.EQ.0)GO TO 500
KK=KK+1
MADJ(I,J)=0
MADJ(J,I)=0
500 CONTINUE
203 WRITE(6,25)
CALL MSG(656,IREL)
CALL SINO(KK,&201,&202,&203)
201 K=0
KX=0
205 CALL MSG(657,IREL)
601 K=K+1
KX=KX+1
IF(K.EQ.KFIM+1)GO TO 600
I=IMIN(KX)
J=JMIN(KX)
IF(MADJ(I,J).EQ.1)GO TO 602
K=K+1
GO TO 601
602 WRITE(6,10)J,I,VMIN(KX)
IF (IREL.EQ.1)WRITE(8,10)J,I,VMIN(KX)
10 FORMAT(10X,'ARCO',I3,'-',I2,3X,F10.2)
IF(K.EQ.KFIM)GO TO 600
IF(K.EQ.15.OR.K.EQ.30)GO TO 603
GO TO 601
603 CALL PAUSA
GO TO 205
600 CALL PAUSA
202 DO 700 I=1,NVER
DO 700 J=1,NVER
700 MADJ(I,J)=MT(I,J)
102 IF(IOPC.EQ.21)CALL NEWOP
RETURN
END
C
C PERMITE A DESORIENTACAO DE UM GRAFO
C
SUBROUTINE DISOR
DIMENSION MADJ(40,40),NSUC(40),VAL(40,40)
```

```
COMMON /BLK1/NERRO,IOPC,IREL,IXIST,ICIRC
COMMON /BLK2/IOR,NVER,NARC,MADJ,NSUC
COMMON /BLK4/IVLR,VAL
IF(IOR.EQ.1)GO TO 102
CALL MSG(659,IREL)
CALL PAUSA
GO TO 103
102 IF(IVLR.EQ.0)GO TO 101
DO 50 I=1,NVER
DO 50 J=1,NVER
IF(MADJ(I,J).EQ.1.AND.MADJ(J,I).EQ.1)VAL(I,J)=AMAX1(VAL(I,
J),VAL(J
*,I))
VAL(J,I)=VAL(I,J)
50 CONTINUE
101 DO 100 I=1,NVER
DO 100 J=1,NVER
MADJ(I,J)=MAX0(MADJ(I,J),MADJ(J,I))
100 CONTINUE
IARC=0
DO 200 I=1,NVER
NSUC(I)=0
DO 200 J=1,NVER
IF(MADJ(I,J).EQ.0)GO TO 200
NSUC(I)=NSUC(I)+1
IARC=IARC+1
200 CONTINUE
NARC=IARC/2
IOR=0
WRITE(6,10)NARC,NVER
IF(IREL.EQ.1)WRITE(6,10)NARC,NVER
10 FORMAT(1H1,///5X,'*** O NOVO GRAFO TEM :',//9X,I6,2X,'ARCO
S',//9X,
+I6,2X,'VERTICES')
CALL MSG(611,IREL)
CALL PAUSA
CALL MENSA
103 IF(IOPC.EQ.22)CALL NEWOP
RETURN
END
```

SUORTE/INICIA

```
C
C   INICIALIZA O ARQUIVO DE GRAFOS ARMAZENADOS
C
FILE 6=SAIDA,UNIT=PRINTER
FILE 15(KIND=DISK,TITLE="SISTEMA/ARQGRA",MAXRECSIZE=1,BLOCKSIZE
      =30)
      IX=0
      DO 100 I=1,5
100  WRITE(15,I)IX
      LOCK(15)
      WRITE(6,10)
10   FORMAT(1H1,///5X,'FOI INICIALIZADO O ARQUIVO',//5X,'NAO EX
      ISTE NEN
      *HUM GRAFO ARMAZENADO')
      STOP
      END
```

SUPORTE/LEITOR

```
C
C   PROGRAMA PARA LER O GRAFO POR CARTOES
C
FILE 5=DADOS,UNIT=READER
FILE 6=SAIDA,UNIT=PRINTER
FILE 15(KIND=DISK,TITLE="SISTEMA/AROGRA",MAXRECSIZE=1,BLOCKSIZE
=30)
SUBROUTINE LER(*,*)
INTEGER SUC(40),MADJ(40,40),NSUC(40)
COMMON /BLK1/IDR,NVER,NSUC,MADJ
DO 100 I=1,NVER
NSUC(I)=0
SUC(I)=0
DO 100 J=1,NVER
100  MADJ(I,J)=0
N=0
150  READ(5,10)I,NSUC(I),(SUC(K),K=1,NSUC(I))
10   FORMAT(20I4)
IF(NSUC(I).EQ.0)GO TO 250
NARC=NARC+NSUC(I)
DO 200 K=1,NSUC(I)
IF(SJC(K).LE.0.OR.SUC(K).GT.NVER)GO TO 350
IF(MADJ(I,SUC(K)).NE.0)GO TO 350
MADJ(I,SUC(K))=1
200  CONTINUE
250  N=N+1
IF(N.NE.NVER)GO TO 150
IF(IDR.EQ.1)RETURN 1
DO 400 I=1,NVER
DO 400 J=1,I
400  MADJ(J,I)=MADJ(I,J)
NARC=0
DO 401 I=1,NVER
NSUC(I)=0
DO 401 J=1,NVER
NSUC(I)=NSUC(I)+MADJ(I,J)
401  NARC=NARC+MADJ(I,J)
NARC=NARC/2
RETURN 1
350  WRITE(6,20)I
20   FORMAT(1H1,///5X,*DADOS INCOERENTES PARA O VERTICE*,I4)
IF((N+1).EQ.NVER)RETURN 2
DO 300 KK=N+2,NVER
300  READ(5)X
RETURN 2
```

END

C
C
C

PROGRAMA PARA GRAVAR O GRAFO LIDO

SUBROJTIME GRAVAR(*)

INTEGER TOP,SUC(40,40),MADJ(40,40),NSUC(40)

COMMON /BLK1/IOR,NVER,NSUC,MADJ

COMMON /BLK2/TOP,MAX

READ(15*1)TOP

READ(15*2)LAST

ITOP=TOP

NAST=LAST

IF(TOP.EQ.0)TOP=1

LAST=LAST+1

CALL PUSH(LAST,&100)

IF(TOP-2)250,150,250

150 CALL PUSH(-1,&100)

CALL PUSH(5,&100)

CALL PUSH(1,&100)

GO TO 250

100 TOP=ITOP

WRITE(6,10)NAST

10 FORMAT(1H1,///5X,*EXCEDEU A CAPACIDADE RESERVADA*,//5X,*UL

TIMO GRA

FO ARMAZENADO NUMERO,I4)

WRITE(6,25)TOP

RETURN 1

250 NEXT=TOP+1

TOP=TOP+1

I=1

550 IF(NSUC(I).EQ.0)GO TO 450

CALL PUSH(I,&100)

CALL PUSH(NSUC(I),&100)

K=0

DO 300 J=1,NVER

IF(MADJ(I,J).EQ.0)GO TO 300

CALL PUSH(J,&100)

K=K+1

SUC(I,K)=J

300 CONTINUE

450 I=I+1

IF(I.LE.NVER)GO TO 550

WRITE(15*1)TOP

WRITE(15*2)LAST

NTOP=TOP+1

```
WRITE(15*'NEXT')NTOP
WRITE(6,20)LAST
20  FORMAT(1H1,///5X,'O GRAFO ARMAZENADO RECEBEU O NUMERO',I4)
CALL SAIDA(SUC,NSUC,NVER)
WRITE(6,25)TOP
25  FORMAT(///5X,'ULTIMA POSICAO GRAVADA',I7)
RETURN
END
```

```
C
C  COLOCA UM ELEMENTO NO ARQUIVO
C
```

```
      SUBROUTINE PUSH(IX,*)
      INTEGER TOP
      COMMON /BLK2/TOP,MAX
      TOP=TOP+1
      IF(TOP.GT.MAX)GO TO 100
      WRITE(15*TOP)IX
      RETURN
100  TOP=TOP-1
      RETURN 1
      END
```

```
C
C  IMPRIME CADA GRAFO ARMAZENADO
C
```

```
      SUBROUTINE SAIDA(SUC,NSUC,NVER)
      INTEGER SUC(40,40),NSUC(40)
      WRITE(6,5)
5     FORMAT(//10X,'LISTA DOS SUCESSORES DOS VERTICES',/)
      DO 100 I=1,NVER
      IF(NSUC(I).EQ.0)GO TO 101
      WRITE(6,10)I,(SUC(I,K),K=1,NSUC(I))
10    FORMAT(/5X,I5,' ..... ',2(T17,20I4,/)
      GO TO 100
101   WRITE(6,11)I
11    FORMAT(/5X,I5,' ..... NAO TEM SUCESSORES')
100   CONTINUE
      RETURN
      END
```

```
C
C  PROGRAMA PRINCIPAL
C
```

```
      INTEGER TOP,MADJ(40,40),NSUC(40)
      COMMON /BLK1/IDR,NVER,NSUC,MADJ
      COMMON /BLK2/TOP,MAX
      N=0
```

```
MAX=10000
150 READ(5,10,END=99)NVER,IOR
10  FORMAT(20I4)
    N=N+1
    IF(NVER.LE.0.OR.NVER.GT.40)GO TO 300
    CALL LER(&100,&200)
100  CALL GRAVAR(&200)
    GO TO 150
200  WRITE(6,20)N
20   FORMAT(/,5X,'O GRAFO NUMERO',I4,' NAO FOI ARMAZENADO')
    GO TO 150
300  WRITE(6,30)N
30   FORMAT(1H1,///5X,'NUMERO DE VERTICES INCOMPATIVEL NO GRAFO
    ',I4,///5
    *X,'ENCERRADO O ARMAZENAMENTO')
99   STOP
    END
```


SUORTE/LISTARQ

```
C
C   IMPRIME O ARQUIVO DE GRAFOS ARMAZENADOS
C
FILE  B=SAIDA,UNIT=PRINTER
FILE  15(KIND=DISK,TITLE="SISTEMA/ARQGRA",MAXRECSIZE=1,BLOCKSIZE
      =30)
      INTEGER TDP,SUC(40,40)
      DIMENSION NSUC(40),LGRA(200)
      N=0
      CALL POP(TDP,N)
      IF(TDP.NE.0)GO TO 101
      WRITE(8,50)
50    FORMAT(1H1,///5X,"NAO EXISTE NENHUM GRAFO ARMAZENADO NO AR
      QUIVO")
      STOP
101   CALL POP(LAST,N)
      KG=0
      N=N+1
      CALL POP(NEXT,N)
250   WRITE(8,10)
10    FORMAT(1H1,///30X,"GRAFOS ARMAZENADOS NO ARQUIVO")
      NVER=0
      NARC=0
      DO 100 I=1,40
100   NSUC(I)=0
      READ(15,NEXT)NGR
      KG=KG+1
      LGRA(KG)=NGR
      READ(15,NEXT+1)NEXT
      N=N+2
150   CALL POP(I,N)
      CALL POP(NSUC(I),N)
      NARC=NARC+NSUC(I)
      DO 200 L=1,NSUC(I)
      CALL POP(J,N)
      IF(J.GT.NVER)NVER=J
      IF(I.GT.NVER)NVER=I
      SUC(L,I)=J
200   CONTINUE
      IF(N.LT.(NEXT-1))GO TO 150
      WRITE(8,20)NGR,NVER,NARC
20    FORMAT(/5X,"GRAFO - NUMERO",I4,3X,I4," VERTICES",I7," A
      RCOS/ARE
      *STAS",/5X,"LISTA DOS SUCESSORES DE CADA VERTICE",/)
      DO 300 I=1,NVER
```

```
IF(NSUC(I).EQ.0)GO TO 301
WRITE(8,30)I,(SUC(L,I),L=1,NSUC(I))
30  FORMAT( /5X,I5,' ..... ',2(T17,20I4,/) )
    GO TO 300
301  WRITE(8,31)I
31   FORMAT(/5X,I5,' .....  NAO TEM SUCESSORES')
300  CONTINUE
    IF(NGR.NE.LAST)GO TO 250
    WRITE(8,40)(LGRA(I),I=1,KG)
40   FORMAT(1H1,///25X,'NUMERO DOS GRAFOS EXISTENTES NO ARQUIVO
      ',//10(5
      *X,20I4,/) )
    WRITE(8,41)LAST,TOP
41   FORMAT(///5X,'ULTIMO GRAFO ARMAZENADO - NUMERO',I4,//5X,'U
      LTIMA PO
      *SICAO GRAVADA',I6,' (MAXIMO = 10000)')
    STOP
    END

C
C   RETIRA UM ELEMENTO DO ARQUIVO
C
SUBROJTINE POP(IX,K)
K=K+1
READ(15,K)IX
RETURN
END
```

SUORTE/APAGA

```
C
C   PROGRAMA PARA APAGAR UM GRAFO DO ARQUIVO
C   ATUALIZA E COMPACTA O ARQUIVO
C
FILE 5=DADOS,UNIT=READER
FILE 6=SAIDA,UNIT=PRINTER
FILE 15(KIND=DISK,TITLE="SISTEMA/ARQGRA",MAXRECSIZE=1,BLOCKSIZE
      =30)
      INTEGER TOP,LIDO(10000)
      READ(15'1)TOP
      READ(15'2)LAST
500   LGRA=0
      IF(TOP.NE.0)GO TO 502
501   WRITE(6,21)
21    FORMAT(1H1,///5X,'NAO EXISTE NENHUM GRAFO NO ARQUIVO')
      STOP
502   READ(5,20,END=99)NGRA
20    FORMAT(I4)
      NEXT=5
450   READ(15'NEXT)IGRA
      IF(IGRA.EQ.NGRA)GO TO 250
      IF(IGRA.EQ.LAST)GO TO 350
      READ(15'NEXT+1)NEXT
      LGRA=IGRA
      GO TO 450
250   K=NEXT
      READ(15'NEXT+1)NEXT
      IDIF=NEXT-K
      IF((NEXT-TOP).EQ.1)GO TO 200
      J=NEXT
504   READ(15'J+1)INEX
      NPUT=INEX-IDIF
      WRITE(15'J+1)NPUT
      IF(INEX.EQ.TOP+1)GO TO 503
      J=INEX
      GO TO 504
503   N=1
      DO 100 I=NEXT, TOP
      READ(15'1)LIDO(N)
      N=N+1
100   CONTINUE
      N=1
      DO 150 I=NEXT, TOP
      WRITE(15'I-IDIF)LIDO(N)
      N=N+1
```

```
150  CONTINUE
      TOP=TOP-IDIF
      GO TO 201
200  TOP=K-1
      WRITE(15,2) LGRA
      LAST=LGRA
      IF(LGRA.EQ.0)TOP=0
201  WRITE(15,1)TOP
      WRITE(6,50)NGRA
50   FORMAT(1H1,///5X,'FOI APAGADO O GRAFO NUMERO',I4,///5X,'O A
      RQUIVO F
      *OI COMPACTADO')
      IF(TOP.EQ.0)GO TO 501
      WRITE(6,60)LAST,TOP
60   FORMAT(/5X,'ULTIMO GRAFO ARMAZENADO - NUMERO',I4,///5X,'ULT
      IMA POSI
      *CAO GRAVADA',I6)
      GO TO 500
350  WRITE(6,90)NGRA
90   FORMAT(1H1,///5X,'O GRAFO NUMERO',I4,' NAO EXISTE NO ARQU
      IVO')
      GO TO 500
99   STOP
      END
```

REFERÊNCIAS

- 1 - Berge, C. - Graphes et Hypergraphes, Dunod, 1970.
- 2 - Berge, C. - Théorie des Graphes et ses Applications, Dunod, 1958.
- 3 - Boaventura Netto, P.O. - Teoria e Modelos de Grafos, Edgard Blücher, 1979.
- 4 - Hu, T.C. - Integer Programming and Network Flows, Addison-Wesley, 1969.
- 5 - Furtado, A.L. - Teoria dos Grafos - Algoritmos, LTC, 1973.
- 6 - Busacker, R.G; Saaty, T.L. - Finite Graphs and Networks: an Introduction with Applications, McGraw-Hill, 1965.
- 7 - Burroughs B6700/B7700, Command and Edit (CANDE) Language, Information Manual, 1972.
- 8 - Pereira Rodrigues, F.C. - Introdução ao uso do B6700, NCE-UFRJ, 1978.
- 9 - Burroughs B7000/B6000 Series - FORTRAN Reference Manual, 1978.
- 10 - Maynard, J. - Programação Modular, LTC, 1977.
- 11 - Pacitti, T; Atkinson, C.P. - Programação e Métodos Computacionais, Vol.1, LTC, 1975.
- 12 - Ford, L.R.; Fulkerson, D.R. - Flows in Networks, Princeton, 1962.
- 13 - Knuth, D.E. - The Art of Computer Programming, Vol.1, Fundamental Algorithms, Addison-Wesley, 1972.
- 14 - IBM System/360 and System/370 FORTRAN IV Language, GC28-6515-9, 1973.
- 15 - IBM, OS/VS2 TSO Command Language Reference, GC28 - 0646, VS2 Release 3, 1975.