

# INSTRUÇÕES DO TRABALHO DE GRAFOS

Leia antes de iniciar

1. INTRODUÇÃO .....	1
2. REQUISITOS BÁSICOS.....	2
3. O QUE DEVE SER ENTREGUE.....	3
4. SOBRE A APRESENTAÇÃO.....	3
5. FORMA DE AVALIAÇÃO.....	4
6. PENALIDADES .....	4
7. MODELO DO RELATÓRIO .....	5
8. TEMAS .....	7

## 1. INTRODUÇÃO

O trabalho consiste em programar um dos temas propostos listados neste documento. Duas equipes diferentes não podem escolher o mesmo tema. O programa deve ler, de um arquivo texto, a matriz de Adjacência de um grafo, (ver layout do arquivo no arquivo exemplo: “exemplo.txt”). O nome do arquivo deve ser digitado pelo usuário, no programa, em tempo de execução.

O grafo contido no arquivo poderá ter de 1 a 100 vértices.

Junto com o arquivo de exemplo, o aluno terá a parte inicial do trabalho já pronta, ou seja, um algoritmo funcionando, que lê o arquivo com o grafo e carrega-o na matriz de Adjacência. O programa deve apresentar na tela os resultados do algoritmo conforme tema escolhido.

Caso o resultado seja um grafo, deve-se exibir a matriz de adjacência do mesmo, com a numeração das linhas e colunas. Não precisa desenhar o grafo.

O programa será apresentado ao professor em data a ser combinada com os alunos. No dia da apresentação deve ser entregue tudo que foi pedido. Ver seção de “O que deve ser entregue”.

## 2. REQUISITOS BÁSICOS

1 – Só pode ser feito em C ou PASCAL. Não serão aceitos em qualquer outra linguagem. NÃO INSISTA.

2 – O código deve estar perfeitamente ENDENTADO, escrito com letras MINÚSCULAS (exceto quando for exigência da linguagem ou para manter as boas práticas de programação) e principalmente, deve ter COMENTÁRIOS bem explicativos sobre toda a lógica implementada. Além disso, deve estar todo em um único arquivo. Pode ser separado em funções e procedimentos, mas tudo em um arquivo único,

3 – A extensão do arquivo não deve ser digitada durante os testes.

4 – O programa não pode fechar após a execução de um arquivo, um novo arquivo deve ser pedido, sem que o programa precise ser reiniciado. O nome do arquivo deverá ser **digitado pelo usuário** em tempo de execução, sem a extensão e o grafo contido nele, poderá ter de 1 a 100 vértices. Qualquer grafo neste intervalo pode ser testado no aplicativo. O arquivo pode ter qualquer nome, **não podendo ser um nome fixo**. Mudanças no código para alterar o nome do arquivo são terminantemente proibidas no momento dos testes. Serão feitos testes com arquivos de vários nomes diferentes.

5 – O rótulo dos vértices do grafo precisam, necessariamente, iniciar em 1. Não pode, de nenhuma forma, existir o “vértice 0” nos grafos a serem testados.

6 – O relatório deve ser entregue em **uma única página**, impresso e em formato digital. Não podendo, por nenhuma hipótese, ter duas ou mais páginas.

7 – No relatório é pedido o cálculo da complexidade. É exigido que as estruturas de repetição/recursão indicativas do cálculo da complexidade estejam claramente identificadas nos COMENTÁRIOS do programa. Para esse cálculo, a parte fornecida pelo professor deve ser considerada.

8 – Para um aluno repetente utilizar o mesmo tema do semestre anterior, é preciso que o trabalho seja feito individualmente.

ATENÇÃO: Qualquer um desses requisitos que não forem atendidos a nota poderá ser pesadamente reduzida, inclusive zerada.

### 3. O QUE DEVE SER ENTREGUE

A entrega destes documentos deve ser efetuada, no dia da apresentação, antes do início da aula em que ocorrerá as apresentações. Qualquer um destes itens que não foi entregue, poderá acarretar perda imediata de 5,0 pontos. Não podendo ser entregue após a apresentação do projeto

- Relatório impresso e em formato digital em uma única página (modelo neste documento).
- Código fonte do programa;
- Executável do programa (com a extensão .EXE);

Devem ser enviados pelo classroom em um único arquivo zipado, **nomeado com o nome do tema**. Não pode ser entregue links para downloads externos nem em arquivos separados. Tem que ser tudo em um único arquivo ZIP.

### 4. SOBRE A APRESENTAÇÃO

1 – Toda a equipe deve apresentar.

2 – A apresentação será apenas para o professor e acontecerá da seguinte forma:

2.1 – Primeiro a equipe entrega o relatório impresso para o professor.

2.2 – Depois eu irei testar alguns grafos trazidos pela própria equipe (tragam pelo menos três e o trivial);

2.3 – Depois eu realizarei perguntas a cada membro da equipe. Enquanto um responde, os demais deverão estar fora da sala e não poderão assistir as apresentações do companheiro de equipe. Qualquer aluno poderá assistir as apresentações das demais equipes que não seja a sua.

3 – A apresentação deverá ser no computador do aluno. Caso nenhum membro da equipe tenha notebook, deverá ser avisado **com antecedência**, para que o professor possa reservar um laboratório para esta apresentação. Neste caso, a equipe deverá necessariamente trazer o trabalho em um pen drive para ser utilizado no laboratório. Não será permitido baixar o trabalho da internet no laboratório. Nenhum aplicativo poderá ser instalado para a apresentação, por isso a exigência do arquivo .EXE do trabalho.

**4 – De nenhuma forma será utilizado o notebook do professor para as apresentações.**

## 5. FORMA DE AVALIAÇÃO

A equipe será avaliada pelos seguintes critérios:

- 5,0 PONTOS: Execução correta do programa apresentando os resultados esperados ao grafo do professor, pelo método correto.
- 4,0 PONTOS: Respostas RÁPIDAS e CORRETAS às perguntas feitas pelo professor Tenorio no momento da apresentação;
- 1,0 PONTO: Entrega do relatório com os dados corretos, inclusive complexidade

## 6. PENALIDADES

1§ Caso o professor constate plágio do algoritmo (cópia da internet, de inteligência artificial ou de outro aluno/equipe de semestres anteriores) em todo ou em parte, todos os alunos envolvidos terão automaticamente nota ZERO, atribuída. Não é considerado plágio, a utilização do código fornecido pelo professor Tenorio, no classroom.

2§ A nota é da equipe, portanto, caso algum aluno da equipe dê respostas erradas às perguntas da apresentação, demore para responde-las, ou responda de forma genérica, TODOS os alunos da equipe perderão pontos.

3§ Alterações feitas no código após a apresentação acarretará em menos 2,0 pontos ao trabalho. Tal penalidade também será aplicada a alterações no relatório ou a qualquer elemento entregue neste projeto.

4§ Caso algum requisito não seja cumprido da seção de requisitos, o aluno perderá a pontuação correspondente.

5§ Caso algum elemento não seja entregue antes da apresentação, da seção “O que deve ser entregue”, a equipe perderá 5,0 pontos.

6§ Caso o relatório seja entregue em mais de uma página, a equipe perderá 1,0 ponto.

**7§ A extensão do arquivo não deve ser digitada durante os testes. Caso precise digitar a extensão, o trabalho terá menos 1,0 ponto.**

**8§ O programa não pode fechar após a execução de um arquivo, um novo arquivo deve ser pedido, sem que o programa precise ser reiniciado. Caso o programa tenha que ser reiniciado para novos testes, o trabalho terá menos 1,0 ponto.**

**9§ Trabalhos feitos em outras linguagens, que não seja C ou PASCAL, terão a nota ZERADA.**

10§ Caso o código não esteja endentado, ou foi escrito com letras maiúsculas, ou com comentários insuficientes, a equipe perderá 4,0 pontos por cada situação destas, sendo um máximo de 10 (dez) pontos.

11§ Caso uma equipe tenha um aluno repetente e o tema escolhido tenha sido o mesmo utilizado pelo repetente, em semestres anteriores, a equipe terá seu trabalho ZERADO.

12§ Caso o aluno não tenha mostrado o trabalho antes da apresentação, nas aulas de acompanhamento, perderá 2,0 pontos.

13§ Aluno repetente, que repetiu tema, precisa mencionar isso no relatório, sob pena de ser considerado plágio de si mesmo, caso não esteja mencionado no relatório.

**14§ Caso o professor precise colocar alguma informação no código fonte para testar novos grafos/arquivos a nota do trabalho será ZERADA.**

**15§ Se o professor precisar alterar alguma coisa no layout do arquivo ou o próprio nome do arquivo para a realização dos testes, a equipe terá a nota ZERADA.**

16§ Terá a nota zerada a equipe que não apresentar o arquivo com a extensão .EXE.

**17§ O rótulo dos vértices tem que iniciar em 1. Será zerada a nota dos trabalhos em que o rótulo dos vértices iniciarem em zero ou em qualquer outro número.**

18§ Equipes com mais componentes do que o máximo permitido terá a nota zerada.

## **7. MODELO DO RELATÓRIO**

A página seguinte possui o modelo do relatório. O relatório deve ser idêntico ao modelo, inclusive em **uma única página** sem adaptações, apenas tendo sido preenchido. O cálculo da complexidade deve ser feito de todo o algoritmo usado, mesmo que tenha utilizado o meu, tem que calcular tudo junto.

**OBS.: Não utilize este modelo no PDF, refaça ele no Word, do zero, para permitir o preenchimento.**

## RELATÓRIO DE TEORIA DOS GRAFOS

Nome:		Matrícula:	
Nome:		Matrícula:	
Nome:		Matrícula:	

Tema			
Nome do arquivo principal do código fonte:		Complexidade:	
Linguagem de Programação usada:			
O que o programa faz?			
Descrição do Algoritmo criado pela equipe (Não colocar códigos, apenas texto)			
Assinatura dos membros da equipe			

## 8. TEMAS

A equipe deve escolher um desses temas. Não pode ser outro e duas equipes não podem escolher o mesmo tema.

MA = Matriz de Adjacência de um Grafo, em alguns casos, pode ser dígrafo.

SN = Deve aparecer na tela a palavra SIM ou NÃO

**ATENÇÃO:** além da saída apresentada no quadro abaixo, precisa **NECESSARIMANTE**, mostrar a matriz de entrada, antes de exibir a resposta. Caso a matriz não esteja presente, a resposta será considerada **ERRADA**.

**ATENÇÃO 2:** A saída não pode ser “em branco” é preciso alguma mensagem, caso o resultado não obtenha nenhum valor.

TEMA	ENTRADA	SAÍDA
Anti-Base	MA Dígrafo	Mostrar os vértices que compõe a anti-base do grafo. Obs.: Todo dígrafo tem anti-base.
Arboricidade	MA	O valor da arboricidade
Articulação	MA	As articulações do grafo. Caso não tenha nenhuma, mostrar a frase: “Não existe articulação neste grafo”.
Árvore de extensão	MA	Quantidade de árvores de extensão.
Árvore de extensão mínima	MA Valorado	Mostrar as arestas que foram excluídas do grafo para que ele se torne árvore de extensão mínima.
Base	MA Dígrafo	Mostrar os vértices que compõe a base do grafo. Obs.: todo dígrafo tem base, mesmo que não tenha fontes.
Bipartido	MA	SN e, em caso positivo, os grupos formados. Não poderá usar o algoritmo de bicoloração neste tema. Laços, se existirem, devem ser desconsiderados.
Caminho alternante.	MA e uma sequência de vértices.	SN (dizer se a sequência de vértices é ou não é um caminho alternante no grafo, em caso positivo, mostrar o emparelhamento utilizado)
Caminho Máximo	MA Valorado, vértice raiz e final.	SN e, em caso positivo, a sequência de arestas do caminho.
Ciclo Euleriano	MA	SN e, em caso positivo, a sequência de arestas que compõe o ciclo.
Ciclo Hamiltoniano	MA	SN e, em caso positivo, a sequência de vértices que compõe o ciclo.
Circunferência e Cintura	MA	SN e a sequência de vértices da circunferência e da cintura.
Clique	MA	Dizer o nome do clique do grafo. Exemplo: K3, K4, etc...
Cobertura Minimal de Arestas	MA	Um conjunto de arestas.

Cobertura Minimal de Vértices	MA	Um conjunto de vértices.
Componentes conexas	MA	A quantidade de componentes conexas e os vértices de cada componente.
ELOS	MA	A quantidade de possíveis Elos do grafo, ou seja, seu posto e quais são as arestas ELOS do grafo.
Emparelhamento	MA	Mostrar um emparelhamento do grafo
Excentricidade	MA	Excentricidade de cada vértice. Não pode usar Dijkstra neste tema.
Fecho Transitivo	MA Dígrafo	Mostrar as setas incluídas para formar o fecho transitivo do grafo.
Forças de Conexão	MA Dígrafo	Dizer se ele é fortemente conexo, unilateralmente conexo, fracamente conexo ou desconexo.
Grafo de arestas	MA	Deve aparecer uma matriz de adjacência do grafo de arestas, com linhas e colunas identificadas.
Grafo Dual	MA	Deve aparecer uma matriz de adjacência do grafo dual, com linhas e colunas identificadas.
Índice cromático	MA	O valor do índice cromático e a cor de cada aresta (as cores são sequências de cor1, cor2, cor3, etc...)
Isomorfismo	2 MA	SN e, em caso positivo, a lista de pares de vértices equivalentes
Número cromático	MA	O valor do número cromático e a cor de cada vértice (as cores são sequências de cor1, cor2, cor3, etc...)
Pancíclico	MA	SN e, em caso positivo, os ciclos usados para identificar que ele é pancíclico. Em caso negativo, quais ciclos faltaram.
Periplanar	MA	SN
Planar	MA com lema 6	SN
Produto lexográfico	2 MA	Deve aparecer uma matriz de adjacência do produto, com linhas e colunas identificadas.
Redução transitiva	MA Dígrafo	Mostrar as setas excluídas para formar a redução transitiva do grafo.
SCEE Minimal	MA Dígrafo	Vértices que compõem o SCEE minimal do dígrafo.
SCIE Maximal	MA	Vértices que compõem TODOS os SCIE maximal do grafo.
Triangulação	MA	SN
Unicursal	MA	SN e a sequência de arestas do caminho e do ciclo euleriano caso existam.

Obs.: quando uma aresta precisar ser mostrada na tela de um grafo sem rótulos, usar a notação (x,y), onde x e y são os vértices que tocam a referida aresta.