

Estudante: _____

Unidade Curricular: Grafos

Professor: Eduardo Hiroshi Nakamura

Curso: Bacharelado em Engenharia de Computação

Campus Três Lagoas

Data ____/____/____



INSTITUTO FEDERAL
MATO GROSSO DO SUL

VALOR 10,0 NOTA ____

ORIENTAÇÕES PARA O TRABALHO

- O nome do arquivo do o código fonte deve obedecer a seguinte sintaxe **nome**_fundamentosgrafos.c, onde o discente deverá substituir a palavra **nome** pelo seu próprio.

OBSERVAÇÕES: Nomes de códigos fontes **não** devem conter espaços e serem grafados somente com caracteres **ASCII** {a a z, A a Z, 0 a 9 e _}.

- Não haverá nota caso falte algum dos itens a seguir: malloc|calloc, free, ponteiro de ponteiro, vetor de ponteiros e lista encadeada(no mínimo a simplesmente encadeada).
- O arquivo entrada do grafo será na sinxtaxe da liguagem **DOT** do Graphviz.

- 1) Criar uma função que verifique se o grafo é conexo.(0.5)
- 2) Criar uma função que verifique se grafo possui loop/laço.(0.5)
- 3) Criar uma função que verifique se grafo possui arestas paralelas.(0.5)
- 4) Criar uma função que calcule o grau de cada vertice.(0.5)
- 5) Criar uma função que calcule o grau mínimo e máximo de um grafo. Imprimir repetições com o idetinficador do vértice.(0.5)
- 6) Criar uma função que calcule a ordem de um grafo.(0.5)
- 7) Criar uma função que verifique se um grafo é simples.(0.5)
- 8) Criar uma função que verifique se um grafo é um multigrafo.(0.5)

- 9) Criar uma função que verifique se um grafo é completo.(0.5)
- 10) Criar uma função que verifique se um grafo é regular.(0.5)
- 11) Criar uma função que caso exista imprima todos os passeios possíveis.(1.0)
- 12) Criar uma função que caso exista imprima todos as trilhas possíveis.(1.0)
- 13) Criar uma função que caso exista imprima todos os circuitos possíveis.(1.0)
- 14) Criar uma função que caso exista imprima todos os caminhos possíveis.(1.0)
- 15) Criar uma função que caso exista imprima todos os ciclos possíveis.(1.0)