

Avaliação 02

Descritivo de Implementação

O presente documento tem por objetivo servir de guia para a elaboração dos trabalhos da SEGUNDA unidade da componente curricular GRAFOS. Ao fim desta unidade, é esperado que o aluno tenha conhecimento sobre a teoria relativa ao conteúdo abordado, e consiga replicá-la em um ambiente computacional.

Como ficou definido na aula inicial, as atividades avaliativas irão constar de implementações computacionais dos pressupostos teóricos vistos em sala de aula. Tais implementações deverão seguir algumas regras básicas.

- (a) As implementações deverão ser feitas em grupo de até 5 participantes (discentes matriculados na turma);
- (b) A linguagem a ser adotada é de livre escolha. Contudo, as implementações devem abranger a teoria vista integralmente. O uso de bibliotecas auxiliares é permitido, porém, não para as atividades fim dos algoritmos. Em outras palavras, as estruturas de dados e pseudocódigos devem ser implementados em sua integralidade;
- (c) Apesar de ser uma atividade em grupo, a avaliação ocorrerá de forma individual, de forma que todos devem ter conhecimento dos códigos e funções implementadas pelo grupo.
- (d) Não custa lembrar que a presença física nas aulas também é objeto de avaliação, contribuindo para a construção da nota individual. Então, evitem faltar.
- (e) A data para entrega (virtual) é o dia **07 de janeiro de 2025**.

Deverá ser entregue:

- i. **O código utilizado, para inspeção:** Os códigos deverão estar comentados, **em português**, detalhando o funcionamento de cada função/procedimento, incluindo o formato da entrada e qual a saída esperada. (Poderá ser disponibilizado em GitHub ou similar)
- ii. **Um vídeo curto**, explicando o funcionamento/estrutura geral do programa, IDE utilizada e os resultados decorrentes da aplicação da função aos dados da avaliação.
- iii. **Uma lista de atividades desenvolvidas por integrante**, detalhando a participação efetiva em cada implementação, seja em sua concepção, implementação, revisão ou teste.

Devido a grande quantidade de conteúdos implementáveis vistos na unidade, os mesmos estão classificados como Obrigatórios ou Opcionais.

Caso o grupo efetue uma implementação DIFERENTE do que foi visto em sala para algum item OBRIGATÓRIO, deverá ser feita a descrição detalhada, a análise de complexidade e análise de corretude do algoritmo implementado. As implementações opcionais são livres.

A implementação de uma função OPCIONAL pode ser feita em **grupo**, onde teremos uma pontuação extra para todos os envolvidos, **ou individual**, com a pontuação sendo considerada apenas para o responsável. Cada implementação OPCIONAL correta acarretará em pontuação extra adicional.

Deverá ser implementado:

A. Para ÁRVORES GERADORES MÍNIMAS

- (1) Algoritmo de Kruskal
- (2) Algoritmo de Prim
- (3) Algoritmo de Boruvka (OPC)
- (4) Algoritmo de Chu-Liu/Edmonds (rodar no exemplo visto em sala)

B. Para CAMINHO MAIS CURTO

- (5) Algoritmo de Dijkstra
- (6) Algoritmo de Bellman-Ford
- (7) Algoritmo de Floyd-Warshall

C. Para GRAFOS EULERIANOS

- (8) Algoritmo de Hierholzer (CICLOS)
- (9) Algoritmo de Hierholzer (CAMINHOS) (OPC)

D. Para FLUXO EM REDES

- (10) Algoritmo de Ford-Fulkerson
- (11) Algoritmo de Edmonds-Karp

Obs. Existem outros algoritmos para cada um dos tópicos estudados. Caso queiram implementar algoritmos diferentes dos que foram vistos, serão analisados como pontuação extra (adicional)

Para o problema de AGM e grafos Eulerianos (Ciclos), considere os seguintes dados:

--	1	2	3	4	5	6	7	8	9	10	11	12
1	0	33	27	34	65	93	145	23	37	23	38	38
2	33	0	28	35	42	94	172	50	64	51	47	65
3	27	28	0	11	69	71	166	44	57	44	72	59
4	34	35	11	0	71	66	174	52	66	53	73	67
5	65	42	69	71	0	192	114	83	97	70	43	94
6	93	94	71	66	192	0	235	112	126	113	133	127
7	145	172	166	174	114	235	0	137	124	133	151	109
8	23	50	44	52	83	112	137	0	14	38	61	38
9	37	64	57	66	97	126	124	14	0	36	67	24
10	23	51	44	53	70	113	133	38	36	0	31	25
11	38	47	72	73	43	133	151	61	67	31	0	56
12	38	65	59	67	94	127	109	38	24	25	56	0

Para o problema de caminho mais curto, considere o seguinte problema ($s = 4$) grafos Eulerianos (Caminhos, início = 1, chegada = 19)

[illegible]

Para o Problema de Fluxo Máximo

