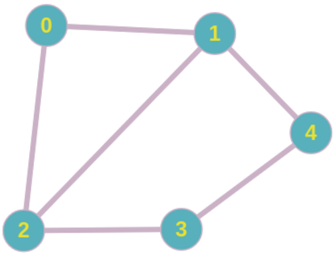
Talles Bezerra de Assunção
Disciplina: Análise de Algoritmos – DCC606

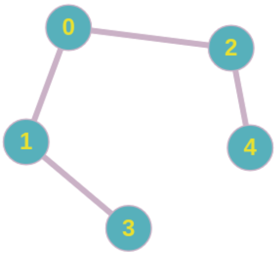
2ª Lista

[QUESTÃO – 02] Defina e dê exemplos:
(A) Grafos.
Um grafo G é um par ordenado G = (V,E) onde, V é o conjunto de vértices e E o conjunto de arestas que conectam esses vértices.

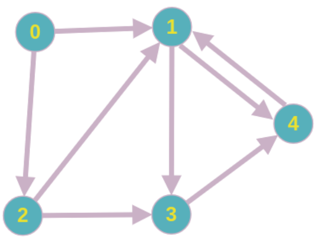


(B) Grafo conexo, acíclico e direcionado.
Grafo conexo consiste em um grafo onde todos os seus vértices estão ligados na mesma cadeia.

No grafo acíclico, para qualquer vértice, não existe um caminho, um ciclo, que retorne para ele.

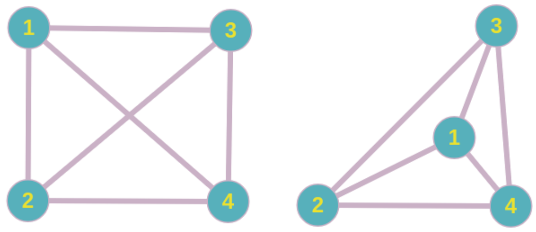


No grafo direcionado, a suas arestas possuem uma direção, por exemplo, na conexão dos vértices u→v, u pode alcançar v, mas v não alcança u.

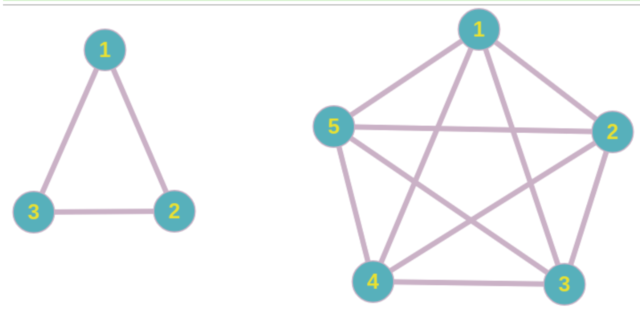


(C) Adjacência x Vizinhança em grafos.
Um vértice é adjacente a outro se existe uma aresta ligada entre eles. Em grafos direcionados, o vértice onde a aresta chega é chamado de sucessor, e o vértice de onde a aresta sai, antecessor.
A vizinhança de um vértice seria o conjuto de vértices ligados a ele.

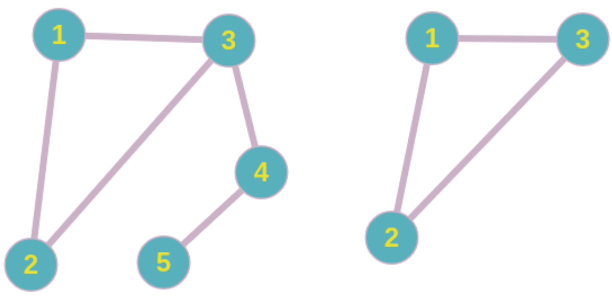
(D) Grafo planar.
Um grafo é planar se for possível organizar os seus vértices de uma forma para que as arestas não se cruzem.



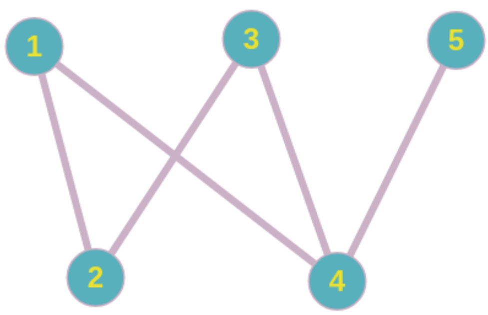
(F) Grafo completo, clique e grafo bipartido.
Um grafo é completo se todos os seus vértices forem adjacentes. Em um grafo completo, o número de arestas é obtido através da fórmula n(n-1)/2, onde n é o número de vértices.
Exemplo:
n = 5
E = 5(5-1)/2 = 20/2 = 10



Um clique de um grafo G é um subgrafo de G, onde esse subgrafo é completo. O clique é denotado por Kn, onde n é o número de vétices do clique.

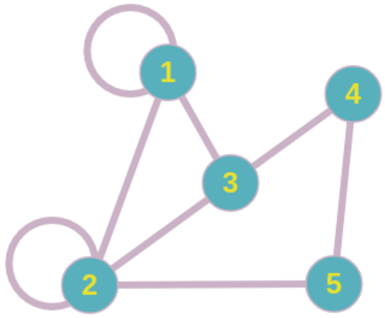


Um grafo é bipartido quando o conjunto de vértices V pode ser particionado em dois subconjuntos V1 e V2, sendo que toda aresta de G une um vértice de V1 a outro de V2.
Exemplo:
V1 = {1,3,5}, V2 = {2,4}



(G) Grafos simples x multigrafo x digrafo.
Um grafo é simples se ele não possuir laços ou arestas múltiplas entre dois vétices.

Um grafo é multigrafo se ele possuir laços ou arestas múltiplas entre dois vétices.



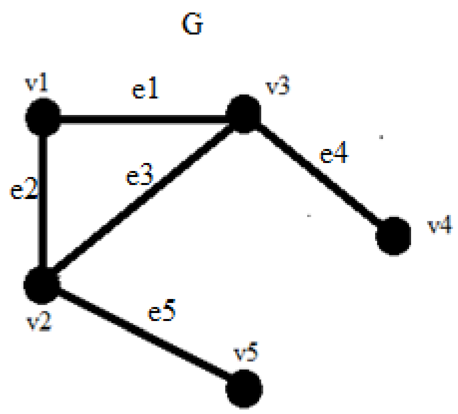
Um digrafo é um grafo em que o seu conjunto de arestas são direcionadas.

[QUESTÃO – 03]
Defina e apresente exemplos de matriz de incidência, matriz de adjacência e lista de adjacência. Adicionalmente, descreva o impacto (vantagens e desvantagens) da utilização de matriz de adjacência e lista de adjacência.

A matriz de incidência Mnxm indica as arestas que estão ligadas aos vétices, onde n é o número de vértices e eles são as linhas, e m é o número de arestas, sendo elas as colunas. Mij recebe 1 quando o vértice i está conectado com a aresta j.

Matriz de incidência de G

	e1	e2	e3	e4	e5
v1	1	1	0	0	0
v2	0	1	1	0	1
v3	1	0	1	1	0
v4	0	0	0	1	0
v5	0	0	0	0	1

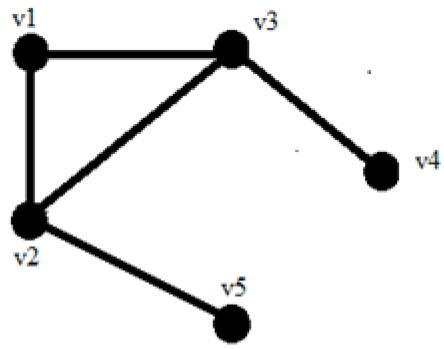


A matriz de adjacência Mnxn indica os vértices que são adjacentes a ele, sendo n o número de vértices do grafo. Mij recebe 1 quando o vértice i é adjacente ao vértice j.

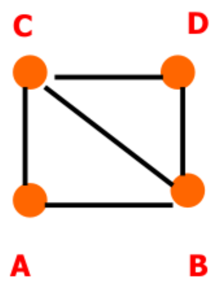
Matriz de Adjacência de G

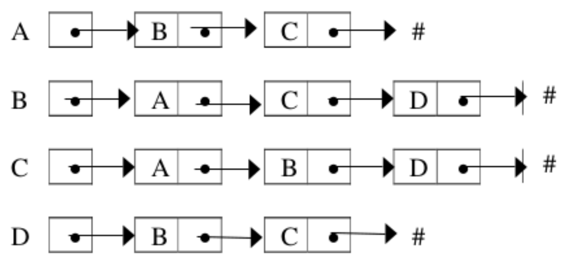
v	1	2	3	4	5
1	0	1	1	0	0
2	1	0	1	0	1
3	1	1	0	1	0
4	0	0	1	0	0
5	0	1	0	0	0

G



A lista de adjacência é um conjunto de n listas A(v), uma para cada vértice v pertencente a V. Cada lista A(v) é denominada lista de adjacências do vértice v e contém os vértices w adjacentes a v em G. Em um grafo direcionado, a lista de adjacência do vértice v possui apenas os vértices sucessores ao v.





A matriz de adjacência possuir vantagens em relação a lista de adjacência quando o grafo que ela representa for não-direcionado e possuir muitas arestas entre os vértices, pois em uma lista teriam que ser adicionados muitos nós para representar todas as arestas dos vértices.
Já para grafos com poucas arestas e direcionados, a lista é uma melhor escolha, pois são adicionados as listas dos vértices somente naqueles que possuir sucessores

[QUESTÃO – 04]
Comente sobre tabelas hash, apresentando a complexidade para as operações realizadas. Adicionalmente, implemente um tabela hash com encadeamento separado usando: lista encadeada e árvore vermelho e preto. Apresente um estudo empírico para obter custos de inserção na medida em que o número de chaves aumenta. Gere gráficos para mostrar o custo de inserção para tamanhos distintos de N (exemplo: de 10 a 10000). Descreva uma análise de comparação em relação ao tempo de execução.

A tabela hash é um vetor que tem como objetivo poder ter acesso a qualquer chave com tempo O(1). O endereço i de acesso da chave k seria obitido atrevés da função de hashing i = k mod n, onde n é a quantidade de posições da tabela que é recomendado que seja um número primo distante de 2.
O algoritmo da tabela hash com lista está na pasta tabela-hash, os arquivos hash.c e hash.h devem estar na mesma pasta para serem compilados. Para verificar diferentes quantidades de números a serem armazenados, alterar a constante LIM_VAL definda no inicio do código
Para compilar, digitar no terminal linux
gcc hash.c -o hash
Executar ./hash

[QUESTÃO – 08]
Defina e exemplifique:
(A) Problema SAT x Teoria da NP-Completude.
O problema SAT consiste em verificar para dada um fórmula booleana formada por AND, OR e NOT, se é possível obter resultado verdadeiro atribuindo valor 0 ou 1 as variaveis dessa fórmula.
Exemplo:
f = (x1 AND x2 AND ~x3) OR (~x1 AND x3), existe valores para x1, x2 e x3 para f ser verdade?
x1 = 0 e x3 = 1.