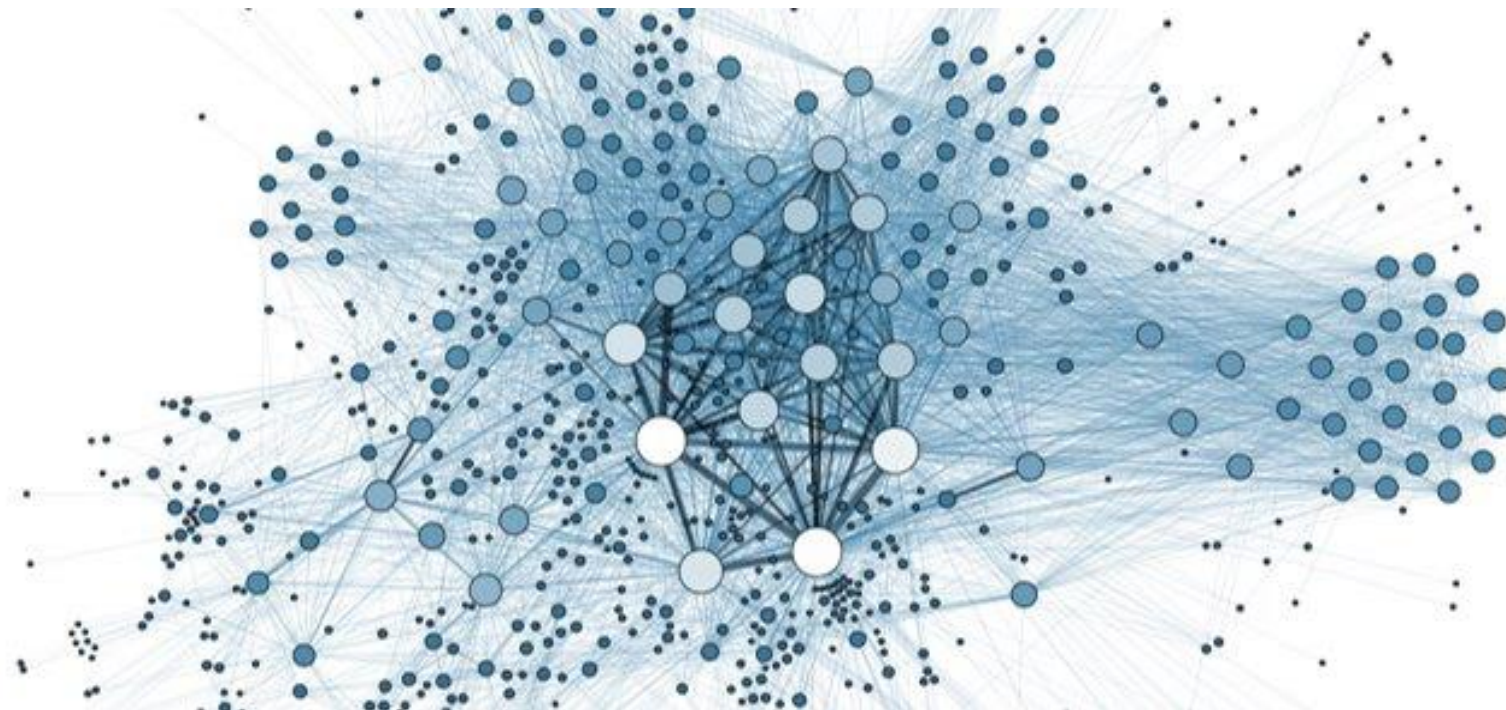


Estrutura de Dados I

Algoritmos em Grafos

Prof. Erinaldo Sanches Nascimento





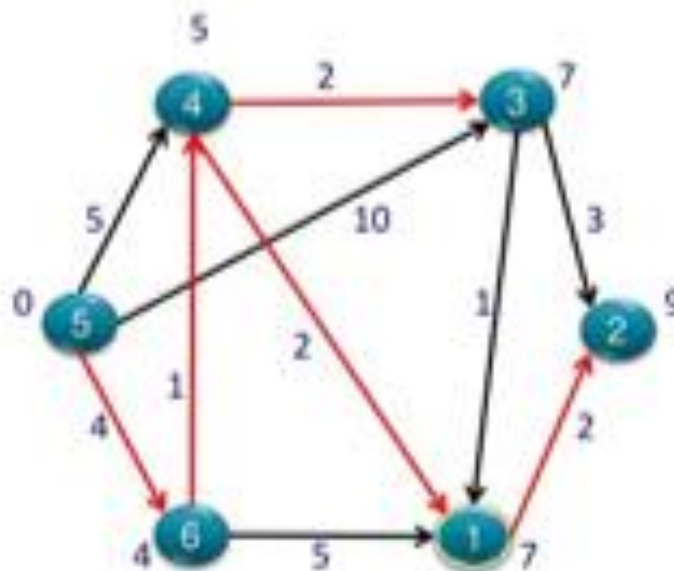
**A teoria dos grafos representa
problemas de conexão entre elementos.**

Ascencio (2010, p. 368)

Unidade 4 e 5

Grafos

1. Teoria dos Grafos
2. Representação Computacional do Grafos
3. Busca em Grafos
4. Algoritmo de Dijkstra

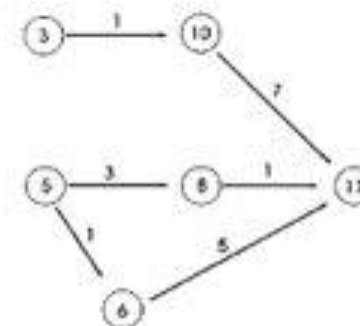
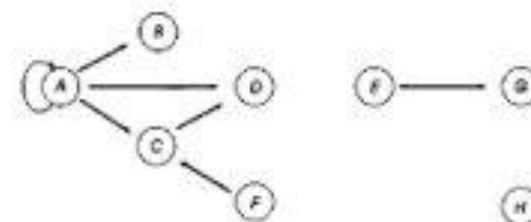
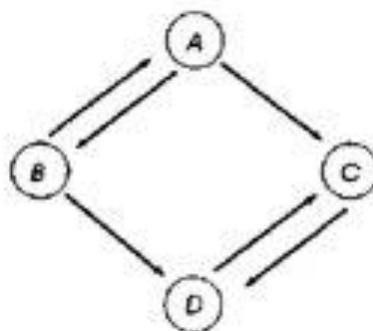
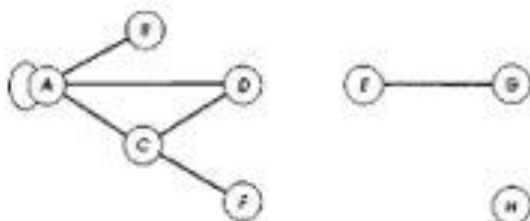
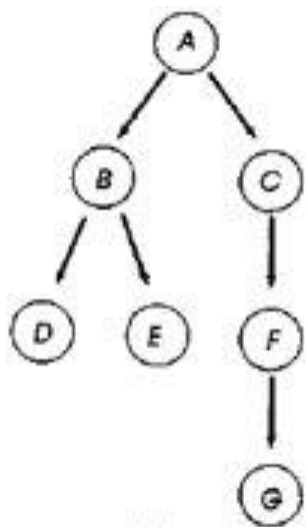


Iter	S	d[1]	d[2]	d[3]	d[4]	d[5]	d[6]
1	0	∞	∞	∞	∞	∞	∞
2	{5}	∞	∞	10	5	0	4
3	{5,6}	9	∞	10	5	0	4
4	{5,6,4}	7	∞	7	5	0	4
5	{5,6,4,1}	7	9	7	5	0	4

Grafos

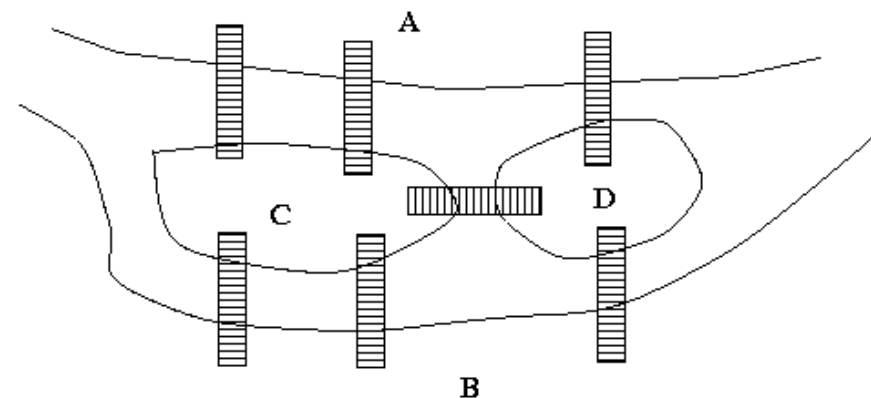
Teoria dos Grafos

Um grafo consiste num conjunto de nós (vértices) e num conjunto de arcos (arestas).



Grafos

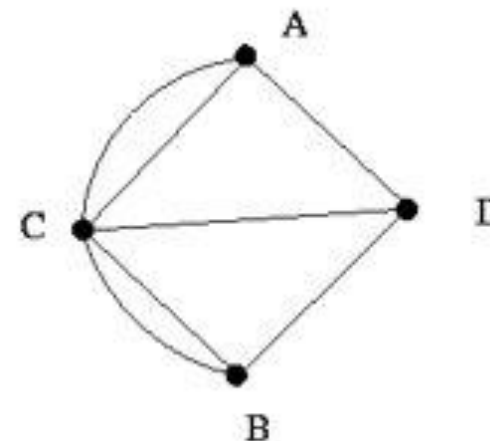
Teoria dos Grafos



As pontes de Königsberg resolvido por Euler.

A estrutura da web pode ser representada por um grafo:

- Quem são os vértices e os arcos?
- O que é navegar na rede?



Grafos

Teoria dos Grafos

- A matriz de adjacências de um grafo é uma matriz de 0's e 1's com colunas e linhas indexadas pelos vértices.
- A matriz de adjacências do grafo cujos arcos são:

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

Grafos

Representação Computacional do Grafos

- Matriz de adjacências
 - Se $adj[][]$ é uma tal matriz então, para cada vértice v e cada vértice w ,
 - $adj[v][w] = 1$ se $v-w$ é um arco e
 - $adj[v][w] = 0$ em caso contrário.
- Listas de adjacências
 - Tem uma lista encadeada associada com cada vértice do grafo.
 - A lista associada com um vértice v contém todos os vizinhos de v .

Vamos implementar?

```
19 main(){
20     while (tamanho <= 0 || tamanho > maximo){
21         tamanho = grafo_tamanho();
22     }
23     if(tamanho <= 0 || tamanho > maximo) {
24         system("cls");
25         printf("Escolha um valor entre 1 e %d!\n\n", maximo);
26     }else {
27         for(int i=0; i<tamanho;i++){
28             grafo[i]=i;
29         }
30     }
```

Grafos

Busca em Grafos

- Busca em profundidade:
 - Visita todos os vértices de um grafo andando pelos arcos de um vértice a outro.
 - O algoritmo de busca em profundidade (*depth-first search*, DFS) visita todos os vértices e numera-os na ordem em que são descobertos.
 - O algoritmo de busca DFS visita todos os vértices e todos os arcos do grafo numa determinada ordem e atribui um número a cada vértice: o k-ésimo vértice descoberto recebe o número k .

Grafos

Busca em Grafos

- Busca em largura
 - Percorre um grafo andando pelos arcos de um vértice a outro.
 - A busca em largura (*breadth-first search*, BFS) está intimamente relacionada com os conceitos de distância e caminho mínimo.
 - A busca em largura começa por um vértice s especificado pelo usuário.
 - O algoritmo visita s , depois visita todos os vizinhos de s , depois todos os vizinhos dos vizinhos, e assim por diante.

Vamos implementar?

```
int main(void) {  
    no *f = (no *) malloc(sizeof(no));  
    if(!f) {  
        printf("Sem memoria disponivel!\n");  
        exit(1);  
    } else {  
        inicia(f);  
    }  
    insere(f);  
    insere(f);  
    insere(f);  
    exhibe(f);  
    retira(f);  
    exhibe(f);  
}
```

Material Complementar

1. Ascencio, A. F. G. **Estrutura de dados:** algoritmos, análise da complexidade e implementações em Java e C/C++. São Paulo: Pearson Prentice Hall, 2010.
2. Tenenbaum, A. M. **Estruturas de dados usando C.** São Paulo: MAKRON Books, 1995.
3. Deitel, P.; Deitel, H. **Java:** Como programar. São Paulo: Pearson Education do Brasil, 2017.



#2022
Realizar

