

Árvores I

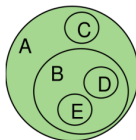
Prof. Kennedy Lopes

UFERSA

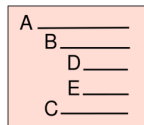
April 14, 2021

Definições e representações básicas

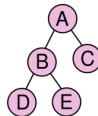
- Uma árvore **T** é um conjunto finito de elementos denominados **nós** ou **vértices** tais que:
 - é **árvore vazia**;
 - possui um valor associado e dois ou mais vértices que são sub-árvores do mesmo formato que T.



c)



b)



a)

1A; 1.1B; 1.1.1D; 1.1.2E; 1.2C

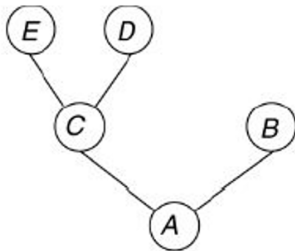
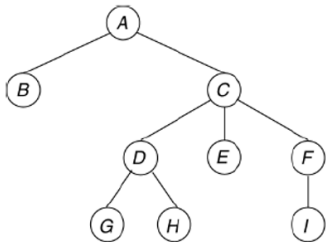
e)

(A(B(D)(E))(C))

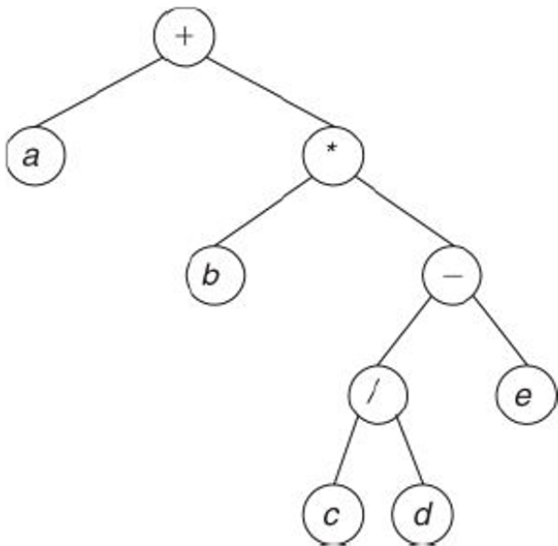
d)

Outras representações

A _____
B _____
C _____
D _____
G _____
H _____
E _____
F _____
I _____



Exemplo



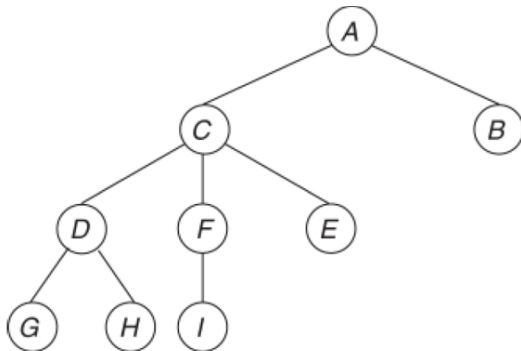
Paretesco em árvores

- É filho de...
- É pai de...

Exemplo:

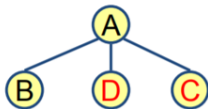
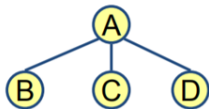
C tem os filhos (D, F, E)

I tem como pai F



Árvores Ordenadas

- Se é considerada a ordem entre os filhos de cada nó, a árvore é **ordenada**.
- **Árvores Isomorfas:** São consideradas isomorfas, árvores nas quais uma permutação de suas sub-árvores as tornam coincidentes;

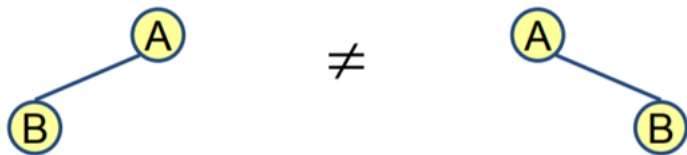


Árvores binárias

Uma árvore binária é uma das duas opções:

- Uma árvore vazia;
- Uma árvore com um nó especial chamado raiz e duas sub-árvores associados a ela:
 - ▶ Sub-árvore esquerda;
 - ▶ Sub-árvore direita.

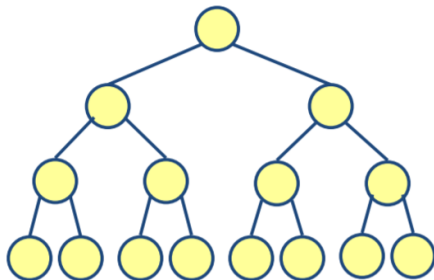
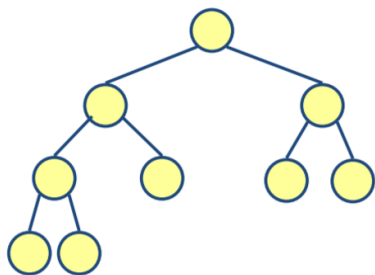
Árvores binárias



Lemma

Uma árvore binária com **n nós**, possui exatamente:

$(n + 1)$ sub-árvores vazias



Tipos especiais de árvores binárias

- Estritamente binária
- Binária completa
- Binária cheia

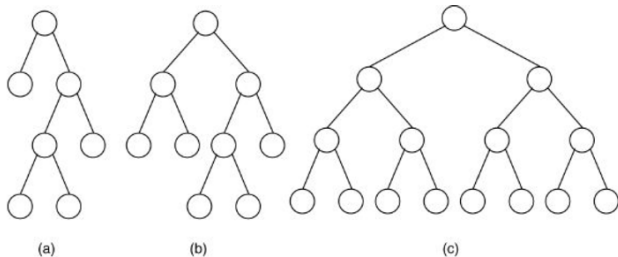
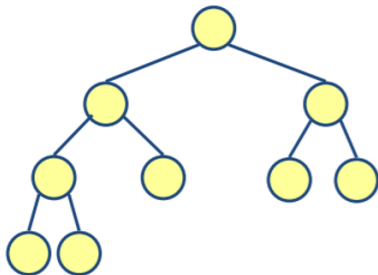


FIGURA 3.7 Árvores estritamente binária, binária completa e cheia.

Altura de árvores binárias

- O processo de busca de dados em árvores
 - Feito a partir da raiz na direção de alguma de suas folhas;
 - Busca ágil: Árvore com a menor altura possível são desejáveis;
- Se uma árvore **T** com $n > 0$ nós é completa, então ela tem altura mínima
 - É possível reordenar folhas para torná-las completas.



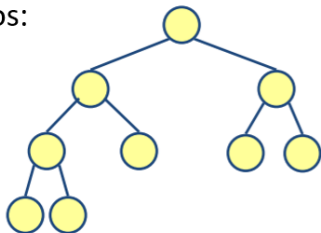
Altura de árvores binárias

- A altura de uma árvore binária pode ser calculada por:

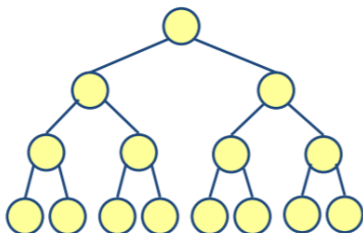
$$h = 1 + \text{piso}(\log_2 n)$$

Onde **piso** representa o maior inteiro menor do que o argumento.

Exemplos:



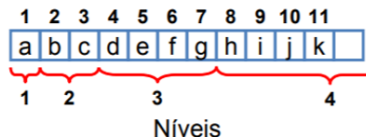
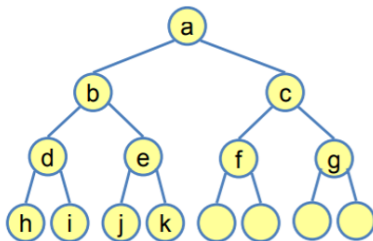
$$h = 1 + \text{floor}[\log_2(9)] = 4$$



$$h = 1 + \text{floor}[\log_2(15)] = 4$$

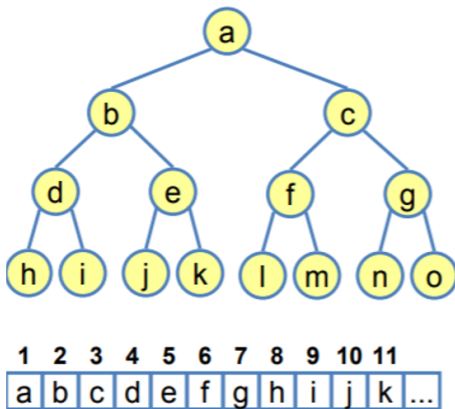
Implementação em forma de vetor

- Árvores binárias implementadas em vetor:
 - Pode-se utilizar o **armazenamento contíguo** proporcionado por vetores;
 - Armazena-se níveis sucessivos da árvore sequencialmente no vetor.



Implementação em forma de vetor

- Considere um nó no índice i . Então seus filhos esquerdos e direitos serão indexados por:
 - Filho esquerdo de i : $2i$
 - Filho direito de i : $2i + 1$
- Uma árvore de altura h precisa de um vetor de $2^h - 1$ elementos;



Implementação com ponteiros


Esq **Val** **Dir**



- Utilizar vetores desperdiça memória;
 - Complicado para reordenar os nós;
 - Ponteiros são mais utilizados.
- Estrutura de dados para um nó da árvore
 - Valor
 - Ponteiro para o nó da esquerda;
 - Ponteiro para o nó da direita.
- A árvore é apenas um ponteiro para o nó raiz.

Implementação com ponteiros

- Estrutura completa da árvore:

- Nó raiz apontado com um ponteiro **T**. 
- Nós folhas apontando para nós sem informação relevante. 