Universidade Federal do Pará
Campus Universitário de Castanhal
Faculdade de Computação
Curso Bacharelado em Sistemas de Informação

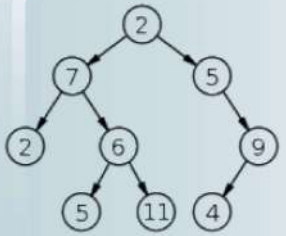
Estrutura de Dados

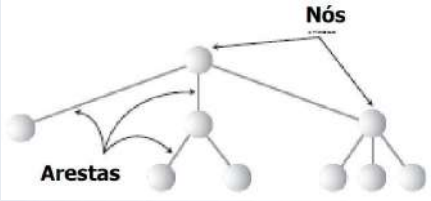
Árvores

Profª Penha Abi Harb
mpenha@ufpa.br

Árvores

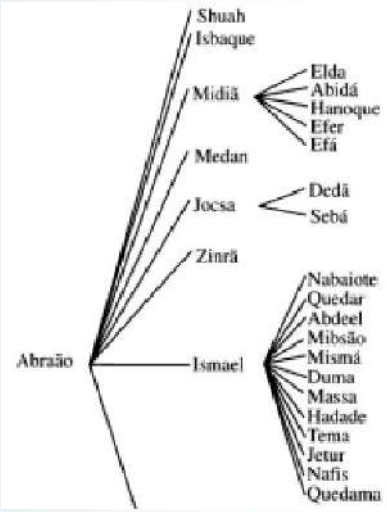
- Com exceção do elemento TOPO, cada elemento da arvore tem um elemento PAI e zero ou mais elementos FILHOS. O elemento TOPO é chamado de RAIZ.
- Consiste em nós conectados por arestas





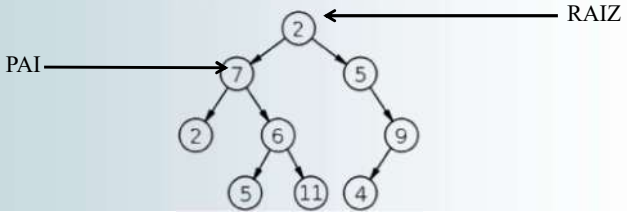
Árvores

- Estrutura de dados não lineares
- Os relacionamentos em uma arvore são hierárquicos, com alguns objetos estando acima e outros abaixo dos outros.
- Ex: arvore genealógica



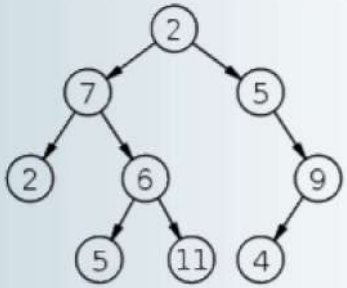
Árvores

- Definição formal: Define-se uma arvore T, como um conjunto de nos que armazenam elementos em relacionamento PAI-FILHO com as seguintes propriedades:
  - Se T não é vazia, ela tem um no especial chamado de raiz de T que não tem pai.
  - Cada no *v* de T diferente da raiz, tem um único no PAI, *w*; todo no com pai *w* é FILHO de *w*



Árvores

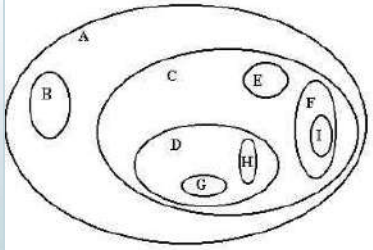
- Dois nos que são filhos do mesmo pai, são irmãos
- Um no *v* é externo se não tem filhos
- Um no *v* é interno se tem um ou mais filhos
- Nos externos também são conhecidos como folha



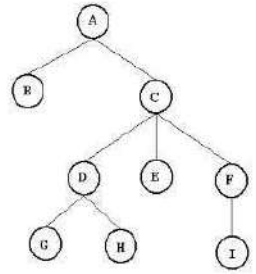
Formas de Representação

Representação por parênteses aninhados
- (A (B) (C (D (G) (H)) (E) (F (I))))

Diagrama de Inclusão

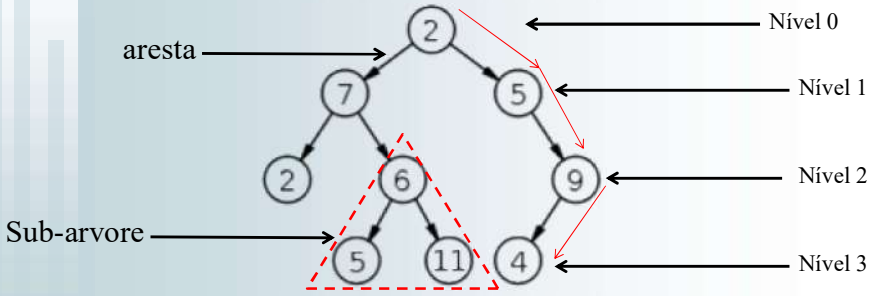


Representação Hierárquica



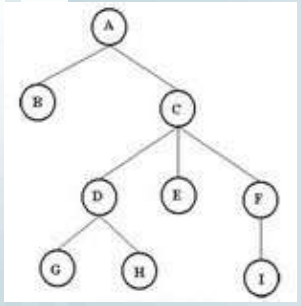
Árvores

- Uma aresta de uma arvore T é um par de nos.
- Um caminho de T é uma sequencia de nos, tais que quaisquer dois nos consecutivos da sequencia formam uma aresta.



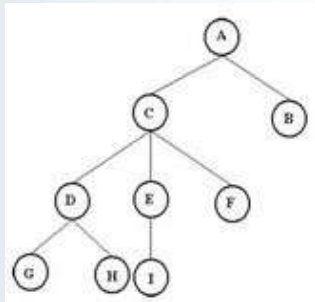
Árvores ordenadas

- É aquela no qual os filhos de cada nó estão ordenados. Assume-se a ordenação da esquerda pra direita.



É ordenada?

Sim

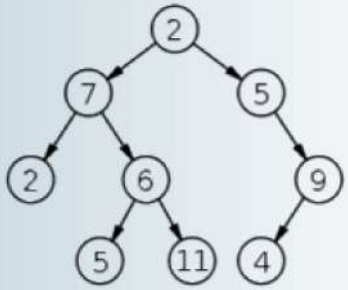


É ordenada?

Não

Árvores ordenadas

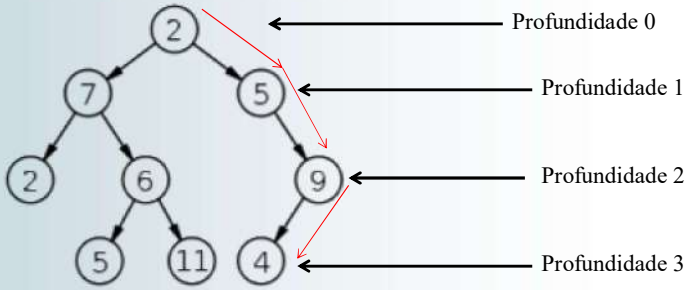
- Se existe uma ordem linear definida entre os filhos de cada no, ou seja, se é possível identificar os filhos como sendo o primeiro, o segundo, o terceiro...



É ordenada?

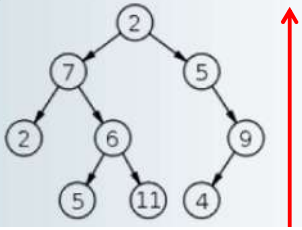
Não

Profundidade



Profundidade

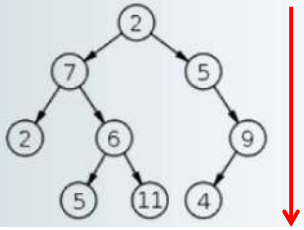
- A profundidade de v, é o numero de ancestrais de v excluindo v, é a distância deste nó até a raiz. Um conjunto de nós com a mesma profundidade é denominado **nível** da árvore.
- Pode ser definida como:
  - Se v é a raiz, a profundidade de v = 0;
  - Em qualquer outro caso, a profundidade de v é um mais a profundidade do pai de v.



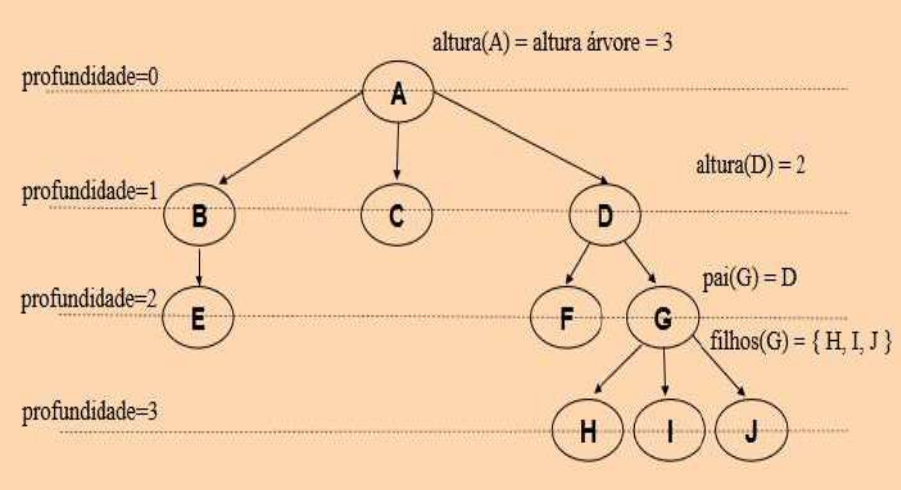
Altura

- A altura de v pode ser definida como:
  - Se v é um no externo, a altura de v = 0;
  - Em qualquer outro caso, a altura de v é um mais a altura máxima dos filhos de v.

A altura de uma arvore T não vazia é igual a profundidade máxima dos nos externos de T.



Profundidade x Altura

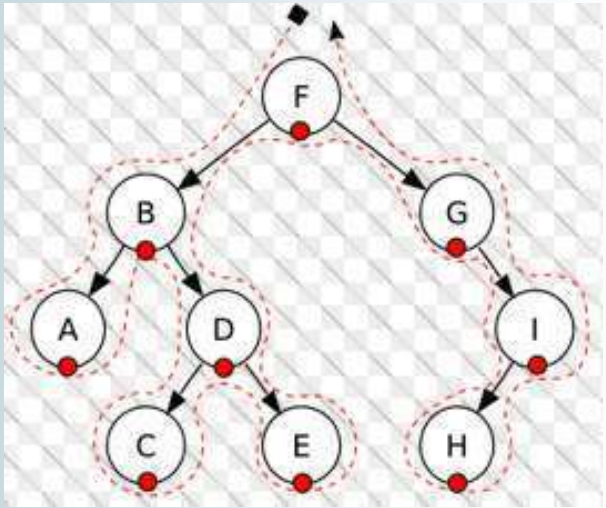


Percorrendo: em ordem

-- Chamar a si mesmo para percorrer a subárvore esquerda o nó
-- Visitar o nó
-- Chamar a si mesmo para percorrer a subárvore direito do nó

Percorrendo

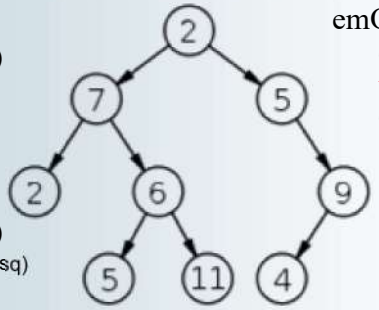
- Visitar cada nó em uma determinada ordem
- Esse processo não é tão usual quanto localizar, inserir e remover.
- Uma razão para isso é que a travessia não é particularmente rápido
- Existem três maneiras de percorrer uma árvore
  - Pré-ordem
  - Em ordem (simétrica)
  - Pós-ordem



percurso : A, B, C, D, E, F, G, H, I.

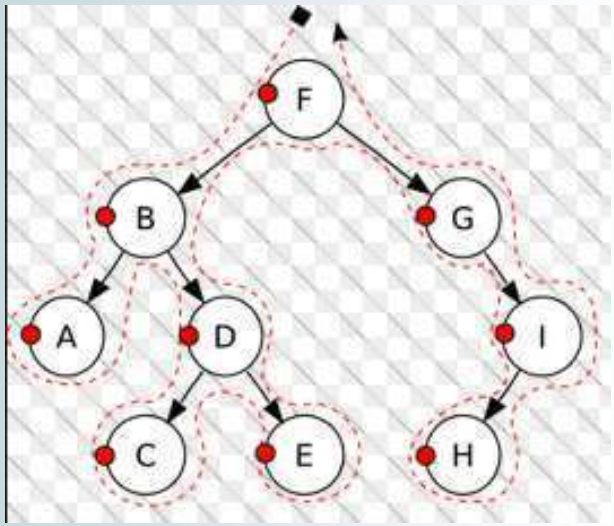
Percorrendo: em ordem

emOrdem(2→esq)
  emOrdem(7→esq)
    emOrdem(2→esq)
    visita(2)
    emOrdem(2→dir)
  visita(7)
  emOrdem(7→dir)
    emOrdem(6→esq)
      emOrdem(5→esq)
      visita(5)
      emOrdem(5→dir)
    visita(6)
    emOrdem(6→dir)
      emOrdem(11→esq)
      visita(11)
      emOrdem(11→dir)



visita(2)
emOrdem(2→dir)
  emOrdem(5→esq)
  visita(5)
  emOrdem(5→dir)
    emOrdem(9→esq)
      emOrdem(4→esq)
      visita(4)
      emOrdem(4→dir)
    visita(9)
    emOrdem(9→dir)

percurso : 2, 7, 5, 6, 11, 2, 5, 4 e 9.



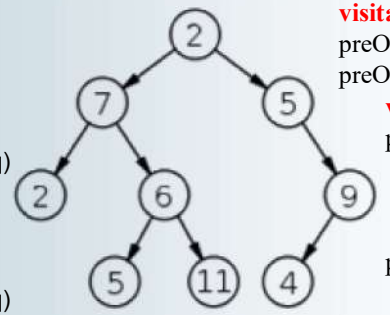
percurso : F, B, A, D, C, E, G, I, H.

Percorrendo: pré-ordem

- Segue os mesmo 3 passos do método EM ORDEM, mas em uma sequencia diferente:
  - Visitar o nó
  - Chamar a si mesmo para percorrer a subárvore esquerda do nó
  - Chamar a si mesmo para percorrer a subárvore direito do nó

Percorrendo: pré-ordem

visita(2)
preOrdem(2→esq)
  visita(7)
  preOrdem(7→esq)
    visita(2)
    preOrdem(2→esq)
    preOrdem(2→dir)
  preOrdem(7→dir)
    visita(6)
    preOrdem(6→esq)
      visita(5)
      preOrdem(5→esq)
      preOrdem(5→dir)
    preOrdem(6→dir)
      visita(11)
      preOrdem(11→esq)
      preOrdem(11→dir)



preOrdem(2→dir)
  visita(5)
  preOrdem(5→esq)
  preOrdem(5→dir)
    visita(9)
    preOrdem(9→esq)
      visita(4)
      preOrdem(4→esq)
      preOrdem(4→dir)
  preOrdem(9→dir)

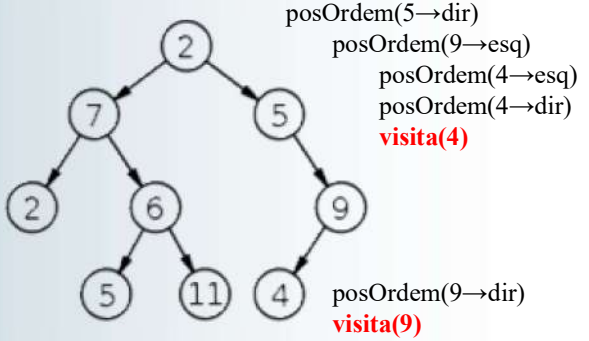
percurso : 2, 7, 2, 6, 5, 11, 5, 9 e 4.

Percorrendo: pós-ordem

- Segue os mesmo 3 passos do método EM ORDEM, mas em uma sequencia diferente:
  - Chamar a si mesmo para percorrer a subárvore esquerda do nó
  - Chamar a si mesmo para percorrer a subárvore direito do nó
  - Visitar o nó

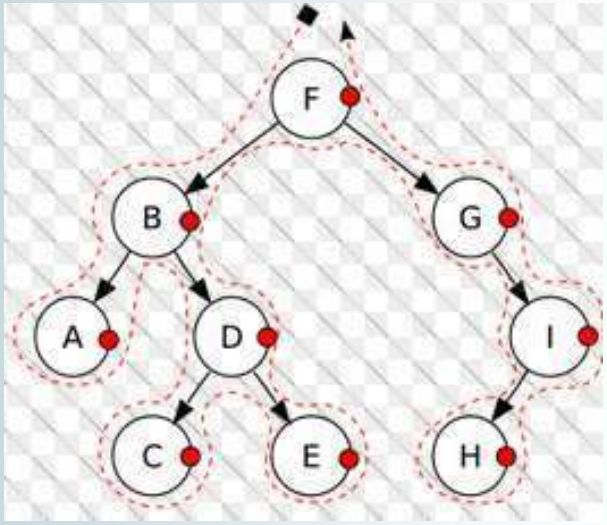
Percorrendo: pós-ordem

posOrdem(2→esq)
posOrdem(7→esq)
posOrdem(2→esq)
posOrdem(2→dir)
visita(2)
posOrdem(7→dir)
posOrdem(6→esq)
posOrdem(5→esq)
posOrdem(5→dir)
visita(5)
posOrdem(6→dir)
posOrdem(11→esq)
posOrdem(11→dir)
visita(11)
visita(6)
visita(7)



posOrdem(2→dir)
posOrdem(5→esq)
posOrdem(5→dir)
posOrdem(9→esq)
posOrdem(4→esq)
posOrdem(4→dir)
visita(4)
posOrdem(9→dir)
visita(9)
visita(5)
visita(2)

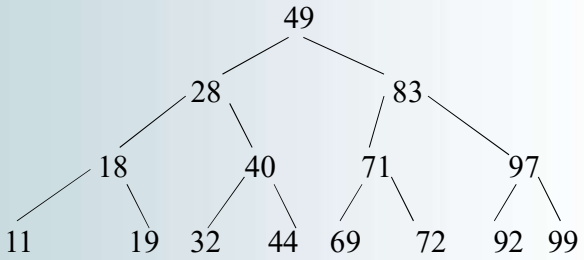
percurso : 2, 5, 11, 6, 7, 4, 9, 5 e 2.



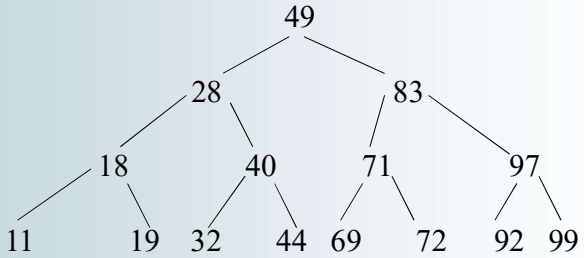
percurso : A, C, E,D ,B, H, I, G, F

Praticando

- Faça manualmente os percurssos na ordem, pré-ordem e pós-ordem da arvore abaixo:



Praticando



Em ordem : esquerda, visita, direita.

Em ordem: 11, 18, 19, 28, 32, 40, 44, 49, 69, 71, 72, 83, 92, 97, 99

Pré-ordem : visita, esquerda, direita.

Pré-ordem: 49, 28, 18, 11, 19, 40, 32, 44, 83, 71, 69, 72, 97, 92, 99

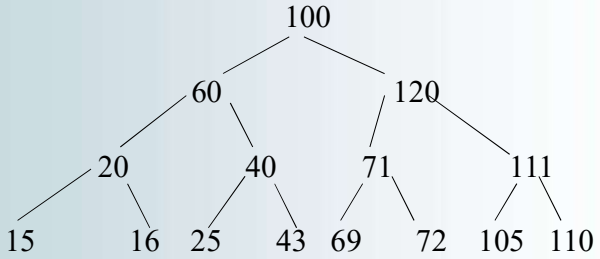
Pós-ordem : esquerda, direita, visita.

Pós-ordem: 11, 19, 18, 32, 44, 40, 28, 69, 72, 71, 92, 99, 97, 83, 49

Arvores binárias

- Uma **árvore binária** é uma estrutura de dados caracterizada por:
  - Ou não tem elemento algum (árvore vazia).
  - Ou tem um elemento distinto, denominado raiz, com dois ponteiros para duas estruturas diferentes, denominadas sub-árvore esquerda e sub-árvore direita.
  - Os nós de uma árvore binária possuem graus zero, um ou dois. Um nó de grau zero é denominado folha.

Praticando 2



Em ordem : esquerda, visita, direita.

Em ordem:

Pré-ordem : visita, esquerda, direita.

Pré-ordem:

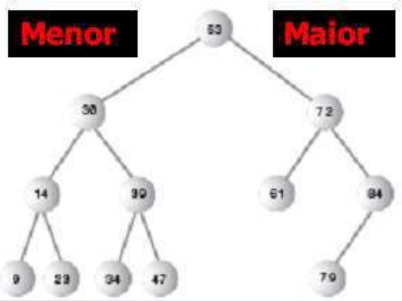
Pós-ordem : esquerda, direita, visita.

Pós-ordem:

Arvores binárias

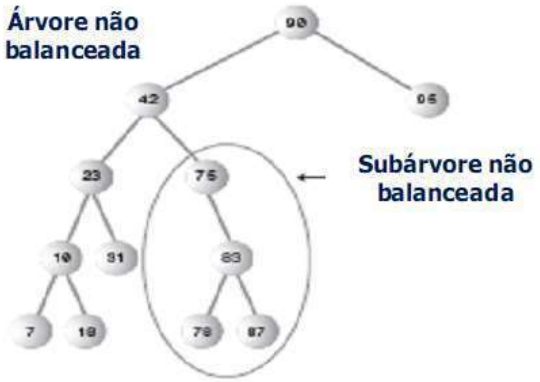
Definição: Todo o nó de uma árvore tiver no máximo dois filhos

- Filho à esquerda
- Filho à direita

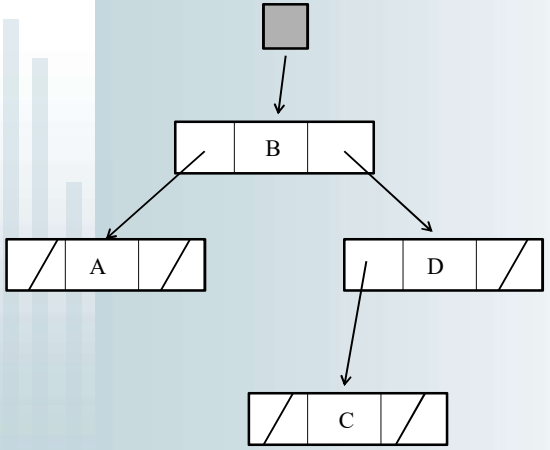


Arvores binárias não balanceadas

Definição: Tem a maioria de seus nós em um lado da raiz ou do outro



Arvores binárias



- Criando os nós:

```
public class no {
    public int valor;

    // Ponteiros para os filhos
    public no filhoesq;
    public no filhodir;
}
```

Arvores binárias e ordenada