

Árvores

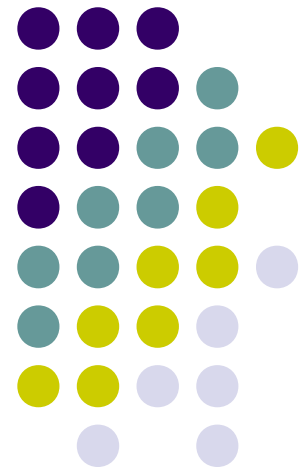
Árvores Genéricas

Árvores Binárias

Rafael Nunes

geocities.yahoo.com.br/rafaelnlp

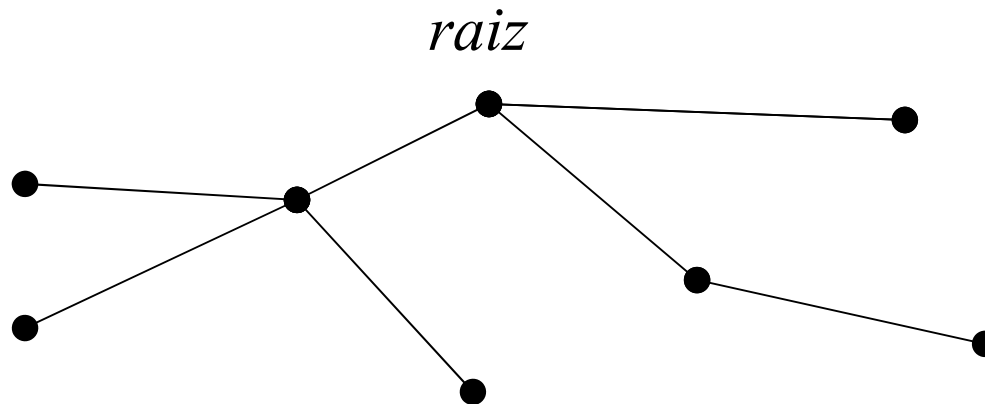
FUMEC – Curso de Ciência da Computação



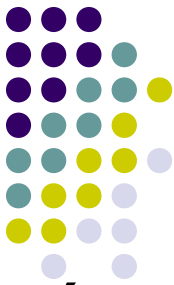
Árvores



- Árvores são estruturas das mais usadas em computação
- Árvores são usadas para representar hierarquias
- Uma árvore pode ser entendida como um grafo acíclico conexo onde um dos vértices – chamado *raiz da árvore* – é diferenciado dos demais

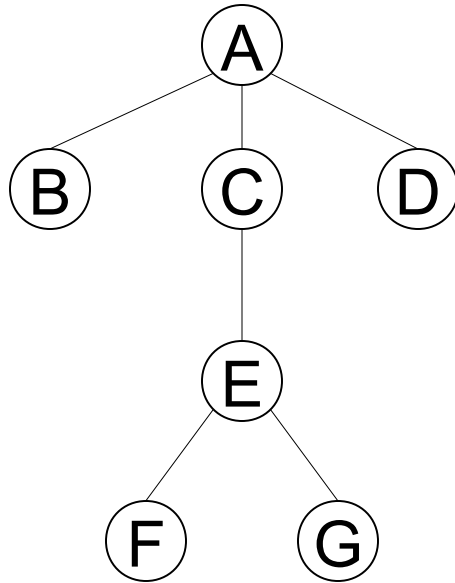


Árvores

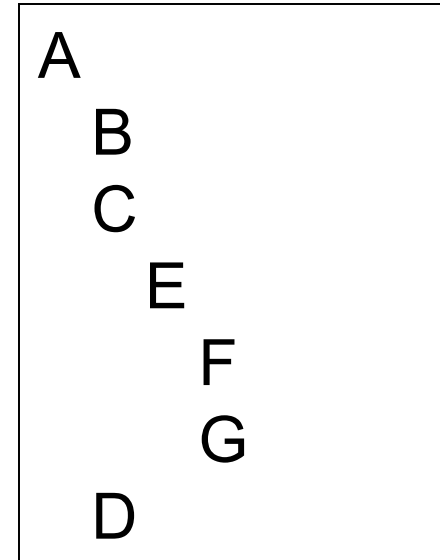


- Uma maneira mais útil de se definir árvores é a seguinte:
 - Uma árvore T é um conjunto finito de nós (ou vértices) tal que
 - $T = \emptyset$, isto é, uma árvore vazia
 - Um nó raiz é um conjunto de árvores não vazias, chamadas de subárvores do nó raiz
- É comum associar-se *rótulos* aos nós das árvores para que possamos nos referir a eles
- Na prática, os nós são usados para guardar informações diversas

Árvores



Representação Gráfica



Representação Indentada

(A(B)(C(E(F)(G)))(D))

Representação com Parênteses

Definições - Nomenclatura



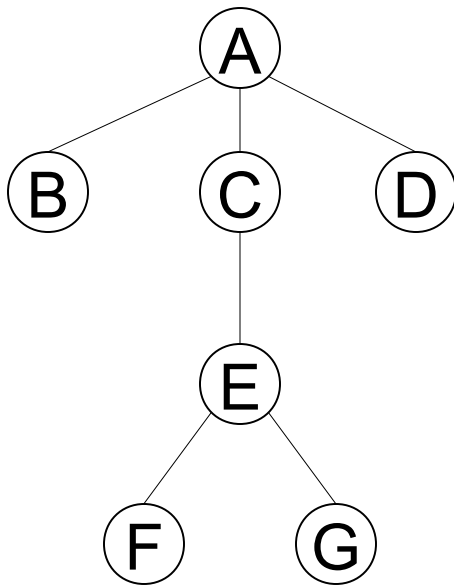
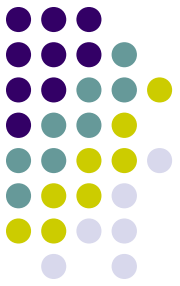
- Dada uma árvore qualquer:
 - A linha que liga dois nodos da árvore denomina-se **aresta**.
 - Diz-se que existe **caminho** entre dois nodos V e W da árvore, se a partir do nodo V puder-se chegar ao nodo W percorrendo-se as arestas que ligam os nodos intermediários entre V e W. Observa-se que existe sempre um caminho entre a raiz e qualquer nodo da árvore.
 - Se houver um caminho entre V e W, começando em V diz-se que V é um nodo **ancestral** de W e W é um nodo **descendente** de V. Se este caminho contiver uma única aresta, diz-se que V é o nodo **pai** de W e que W é um nodo **filho** de V. Dois nodos que são nodos filhos do mesmo nodo pai são denominados nodos **irmãos**. Uma característica inerente a árvores é que qualquer nodo, exceto a raiz, ***tem um único nodo pai.***

Definições - Nomenclatura



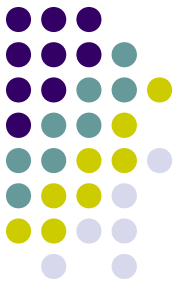
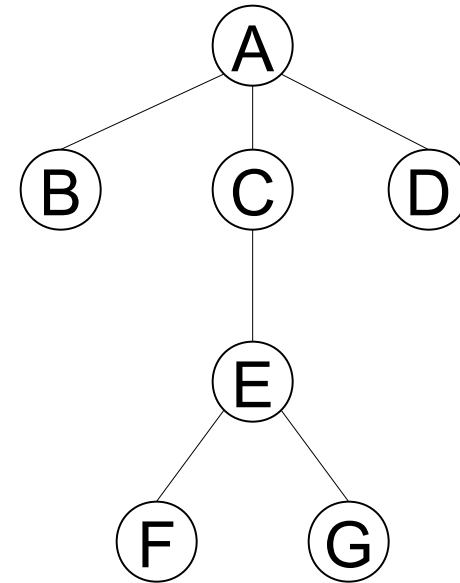
- Dada uma árvore qualquer:
 - Se um nodo não possui nodos descendentes, ele é chamado de **folha** ou nodo **terminal** da árvore.
 - **Grau** de um nodo é o número de nodos filhos do mesmo. Obviamente que um nodo folha tem grau zero.
 - **Nível** de um nodo é o número de nodos existentes no caminho entre a raiz e o próprio nodo. A raiz tem nível 1.
 - O **grau** da árvore é igual ao grau do nodo de **maior grau da árvore**.
 - O **nível** da árvore é igual ao nível do nodo de **maior nível da árvore**.

Árvores – Nomenclatura



- “A” é o pai de “B”, “C” e “D”
- “B”, “C” e “D” são filhos de “A”
- “B”, “C” e “D” são irmãos
- “A” é um ancestral de “G”
- “G” é um descendente de “A”
- “B”, “D”, “F” e “G” são nós folhas
- “A”, “C” e “E” são nós internos
- O grau do nó “A” é 3
- O comprimento do caminho entre “C” e “G” é 2
- O nível de “A” é 1 e o de “G” é 4
- A altura da árvore é 4

Árvores – Exercício

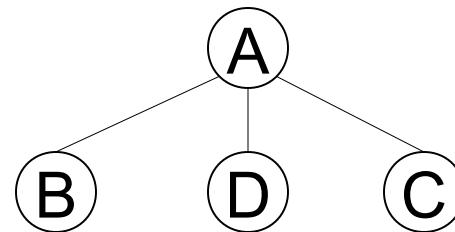
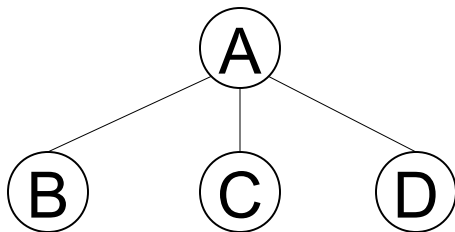


- a) Qual é a raiz da árvore?
- b) Quais são os nodos terminais?
- c) Qual o grau da árvore?
- d) Qual o nível da árvore?
- e) Quais são os nodos descendentes do nodo D ?
- f) Quais são os nodos ancestrais do nodo # ?
- g) Os nodos 4 e 5 são nodos irmãos?
- h) Há caminho entre os nodos C e S?
- i) Qual o nível do nodo 5?
- j) Qual o grau do nodo A?

Árvores Ordenadas



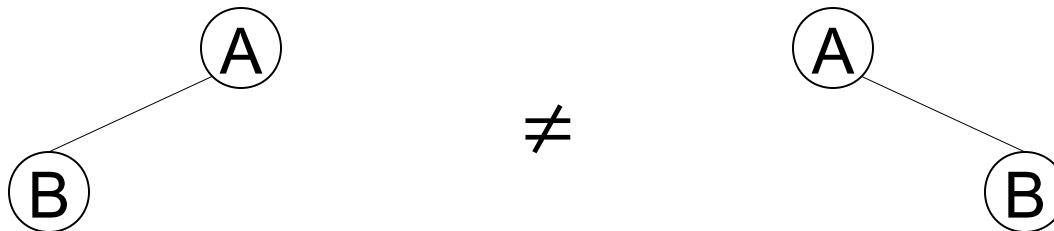
- Se é considerada a ordem entre os filhos de cada nó, a árvore é chamada de *ordenada*
- Pode-se definir o conceito de árvores isomorfas quando elas têm a mesma relação de incidência entre nós mas são desenhadas de forma diferente, isto é, são distintas quando consideradas como árvores ordenadas



Árvores Binárias



- Uma *árvore binária* é
 - Uma árvore vazia ou
 - Um nó *raiz* e duas subárvores binárias denominadas subárvore *direita* e subárvore *esquerda*
- Observe que uma árvore binária não é propriamente uma árvore já que os filhos de cada nó têm nomes (esquerdo e direito)



Número de Subárvores Vazias

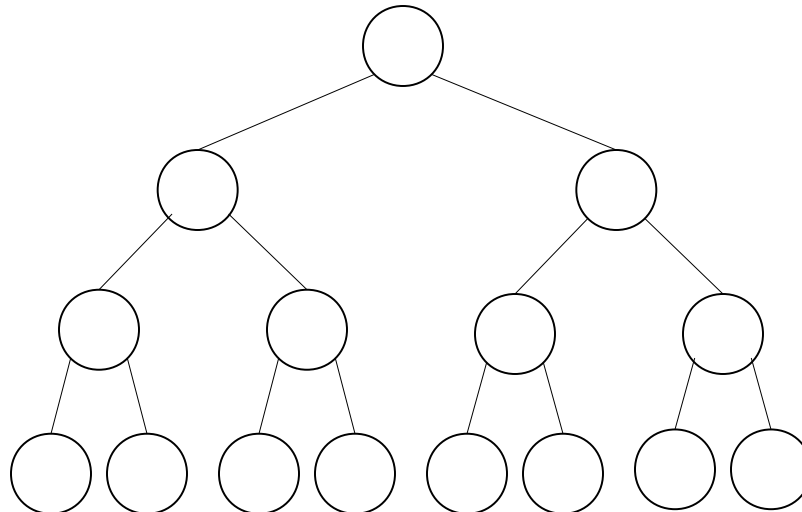


- Se uma árvore tem $n > 0$ nós, então ela possui $n+1$ subárvores vazias
- Para ver isso, observe que
 - Uma árvore com um só nó tem 2 subárvores vazias
 - Sempre que “penduramos” um novo nó numa árvore, o número de nós cresce de 1 e o de subárvores vazias também cresce de 1

Tipos Especiais de Árvores Binárias



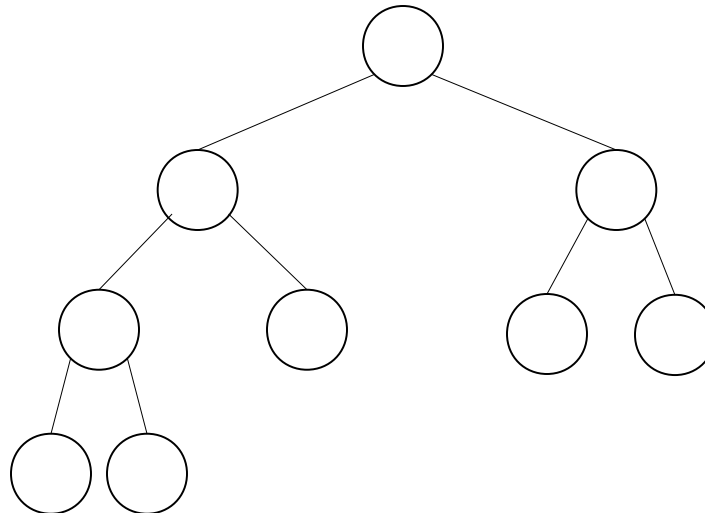
- Uma árvore binária é estritamente binária sse todos os seus nós têm 0 ou 2 filhos
- Uma árvore binária completa é aquela em que todas as subárvores vazias são filhas de nós do último ou penúltimo nível
- Uma árvore binária cheia é aquela em que todas as subárvores vazias são filhas de nós do último nível



Altura de Árvores Binárias



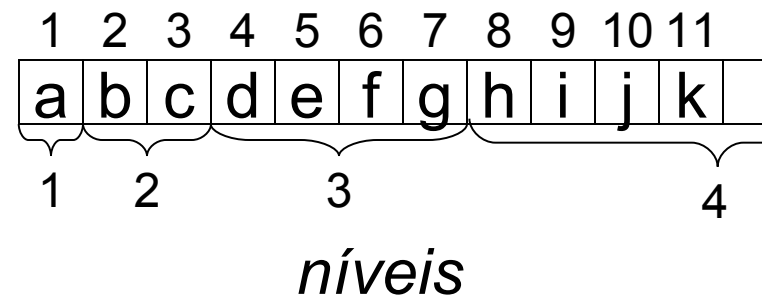
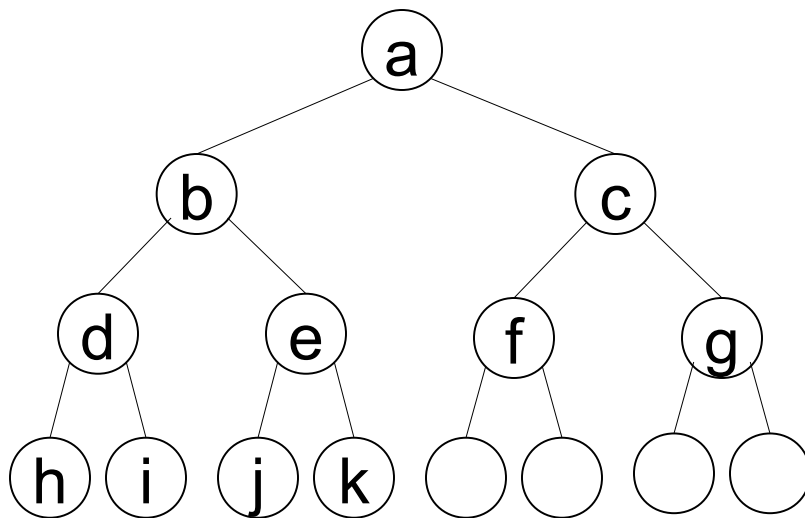
- O processo de busca em árvores é normalmente feito a partir da raiz na direção de alguma de suas folhas
- Naturalmente, são de especial interesse as árvores com a menor altura possível
- Se uma árvore T com $n > 0$ nós é completa, então ela tem altura mínima. Para ver isso observe que mesmo que uma árvore mínima não seja completa é possível torná-la completa movendo folhas para níveis mais altos



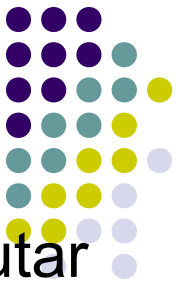
Implementando Árvores Binárias com Arrays



- Assim como listas, árvores binárias podem ser implementadas utilizando-se o armazenamento contíguo proporcionado por arrays
- A idéia é armazenar níveis sucessivos da árvore seqüencialmente no array



Implementando Árvores Binárias com Arrays

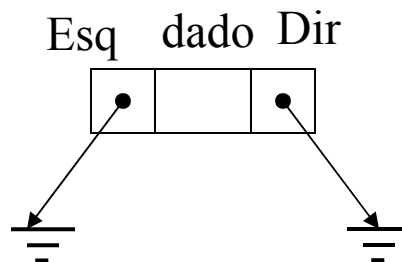


- Dado um nó armazenado no índice i , é possível computar o índice
 - do nó filho esquerdo de i : $2i$
 - do nó filho direito de i : $2i + 1$
 - do nó pai de i : $i \text{ div } 2$
- Para armazenar uma árvore de altura h precisamos de um array de $2^h - 1$ (número de nós de uma árvore cheia de altura h)
- Nós correspondentes a subárvores vazias precisam ser marcados com um valor especial diferente de qualquer valor armazenado na árvore
- A cada índice computado é preciso se certificar que está dentro do intervalo permitido
 - Ex.: O nó raiz é armazenado no índice 1 e o índice computado para o seu pai é 0

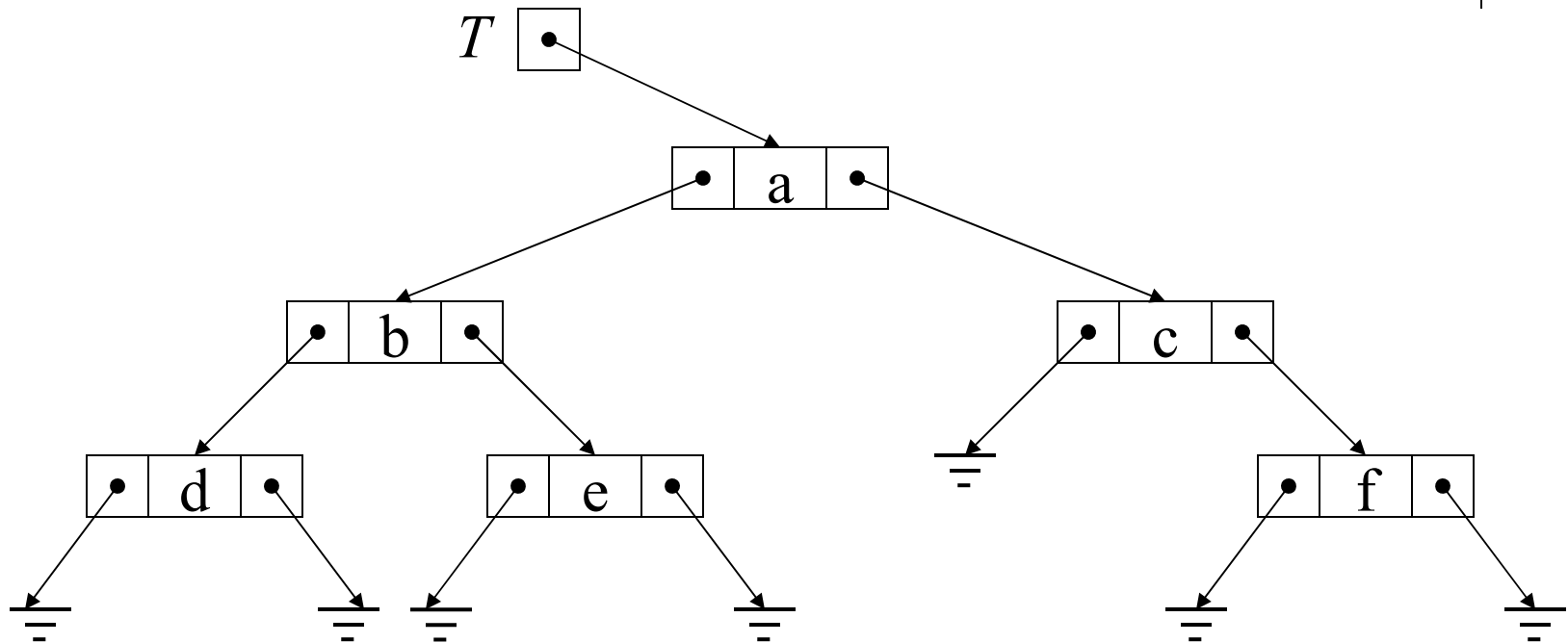
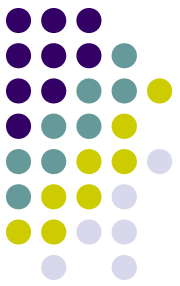
Implementando Árvores Binárias com Ponteiros



- A implementação com arrays é simples porém tende a desperdiçar memória, e é pouco flexível quando se quer alterar a árvore (inserção e deleção de nós)
- Via-de-regra, árvores são implementadas com ponteiros:
 - Cada nó X contém 3 campos:
 - $X.Val$: valor armazenado no nó
 - $X.Esq$: Ponteiro p/ árvore esquerda
 - $X.Dir$: Ponteiro p/ árvore direita
 - Uma árvore é representada por um ponteiro para seu nó raiz



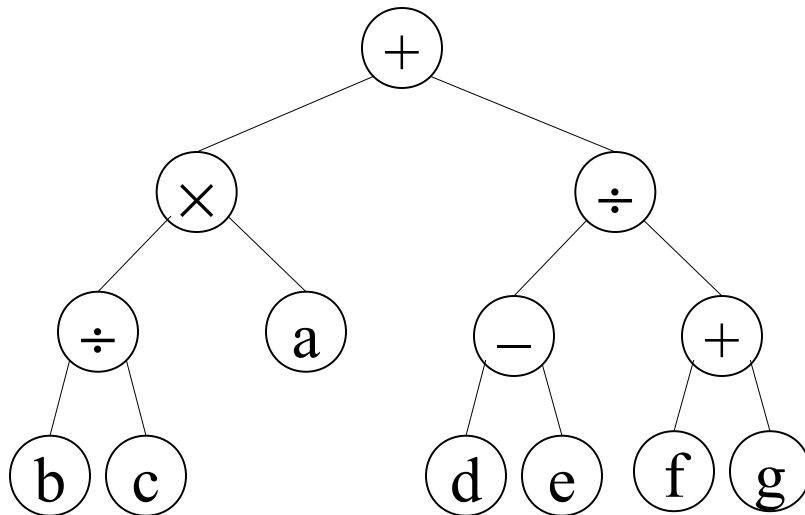
Implementando Árvores Binárias com Ponteiros



Aplicação: Expressões

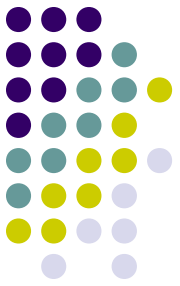


- Uma aplicação bastante corriqueira de árvores binárias é na representação e processamento de expressões algébricas, booleanas, etc



$((b/c) * a) + ((d-e)/(f+g))$

Exercício



- Desenhe a árvore binária que represente a expressão abaixo
- $(((10 / 2) * 3) - 10) + ((26 - 2) / (6 + 2)))$