

MC202 - Estrutura de Dados

Alexandre Xavier Falcão

Instituto de Computação - UNICAMP

afalcao@ic.unicamp.br

Árvore Binária

Uma árvore binária é um conjunto finito T de objetos, denominados nós ou vértices, tal que:

Árvore Binária

Uma árvore binária é um conjunto finito T de objetos, denominados nós ou vértices, tal que:

- ou T é vazio, ou

Árvore Binária

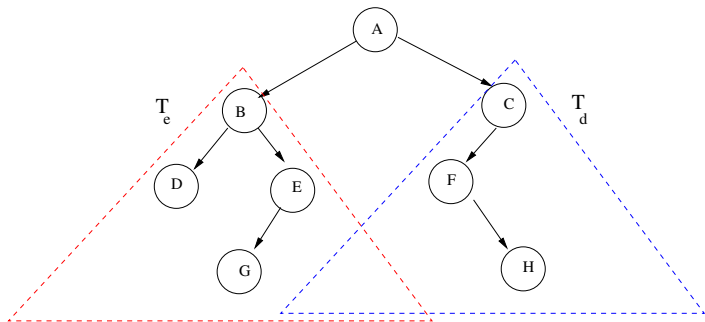
Uma árvore binária é um conjunto finito T de objetos, denominados nós ou vértices, tal que:

- ou T é vazio, ou
- T consiste de um nó distinto, denominado raiz de T , e de duas árvores binárias disjuntas, T_e e T_d , denominadas subárvores esquerda e direita de T (ou da raiz de T), respectivamente. Esta definição é recursiva para qualquer nó.

Árvore Binária

Uma árvore binária é um conjunto finito T de objetos, denominados nós ou vértices, tal que:

- ou T é vazio, ou
- T consiste de um nó distinto, denominado raiz de T , e de duas árvores binárias disjuntas, T_e e T_d , denominadas subárvores esquerda e direita de T (ou da raiz de T), respectivamente. Esta definição é recursiva para qualquer nó.



- Conceitos relacionados.
- Propriedades.
- Percursos.
- Construção de uma árvore binária.

Conceitos relacionados

- O grau de um nó é o número de subárvores não vazias do nó. O nó A tem grau 2, o D tem grau 0, e o F tem grau 1.

Conceitos relacionados

- O grau de um nó é o número de subárvores não vazias do nó. O nó A tem grau 2, o D tem grau 0, e o F tem grau 1.
- Nós de grau 0 são denominados terminais ou folhas (D , G , e H). Os demais são internos ou não-terminais.

Conceitos relacionados

- O grau de um nó é o número de subárvores não vazias do nó. O nó A tem grau 2, o D tem grau 0, e o F tem grau 1.
- Nós de grau 0 são denominados terminais ou folhas (D , G , e H). Os demais são internos ou não-terminais.
- Cada nó é denominado pai dos nós raízes de suas subárvores, e estes são denominados seus filhos à esquerda e à direita. O nó C é pai de F , seu único filho à esquerda. Os nós D e E são irmãos, ambos filhos de B .

- A raiz da árvore está no nível 1. Se um nó está no nível n , seus filhos estão no nível $n + 1$, $n \geq 1$. O nó A está no nível 1, os nós B e C estão no nível 2, os nós D , E , e F estão no nível 3, e os nós G e H estão no nível 4.

- A raiz da árvore está no nível 1. Se um nó está no nível n , seus filhos estão no nível $n + 1$, $n \geq 1$. O nó A está no nível 1, os nós B e C estão no nível 2, os nós D , E , e F estão no nível 3, e os nós G e H estão no nível 4.
- A altura (ou profundidade) de uma árvore é o nível máximo atribuído a um de seus nós. No caso a altura da árvore do exemplo é 4. A exceção é a árvore vazia, cuja altura é 0.

- Sem contar com o apontador para a raiz, uma árvore com n nós utiliza $n - 1$ apontadores não-nulos.

- Sem contar com o apontador para a raiz, uma árvore com n nós utiliza $n - 1$ apontadores não-nulos.
- O número máximo de nós em uma árvore de altura h é $2^h - 1$ (isto ocorre quando a árvore está cheia!).

- Sem contar com o apontador para a raiz, uma árvore com n nós utiliza $n - 1$ apontadores não-nulos.
- O número máximo de nós em uma árvore de altura h é $2^h - 1$ (isto ocorre quando a árvore está cheia!).
- No pior caso para a busca de um elemento, uma árvore binária com n nós é dita degenerada em uma lista ligada, tendo altura $h_{\max} = n$. Neste caso a busca terá complexidade $O(n)$.

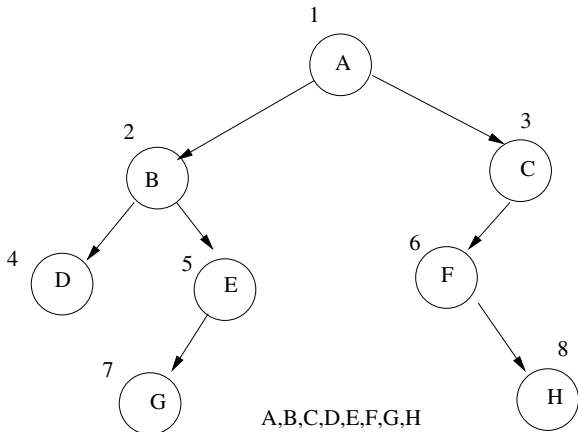
- Sem contar com o apontador para a raiz, uma árvore com n nós utiliza $n - 1$ apontadores não-nulos.
- O número máximo de nós em uma árvore de altura h é $2^h - 1$ (isto ocorre quando a árvore está cheia!).
- No pior caso para a busca de um elemento, uma árvore binária com n nós é dita degenerada em uma lista ligada, tendo altura $h_{\max} = n$. Neste caso a busca terá complexidade $O(n)$.
- No melhor caso para a busca de um elemento, uma árvore binária com n nós tem altura $h_{\min} = \lceil \log_2^{(n+1)} \rceil$. Neste caso, a complexidade da busca será $O(\log_2^n)$.

Percurso em largura

Os nós da árvore são visitados em largura quando o percurso segue a ordem crescente de nível dos nós.

Percurso em largura

Os nós da árvore são visitados em largura quando o percurso segue a ordem crescente de nível dos nós.



Este percurso pode ser facilmente implementado com ajuda de uma fila.

- 1 Insere a raiz em uma fila.
- 2 Enquanto a fila não estiver vazia, faça
- 3 Remova nó p da fila.
- 4 Imprime informação de p .
- 5 Se sua árvore esquerda não for vazia, insere sua raiz na fila.
- 6 Se sua árvore direita não for vazia, insere sua raiz na fila.

Os nós de uma árvore podem ser visitados em profundidade (i.e., seguindo a definição recursiva de árvore binária). Neste caso, existem três alternativas padrões:

Os nós de uma árvore podem ser visitados em profundidade (i.e., seguindo a definição recursiva de árvore binária). Neste caso, existem três alternativas padrões:

- Visita em pré-ordem.
- Visita em ordem simétrica (in-order)
- Visita em pós-ordem.

Percurso em pré-ordem

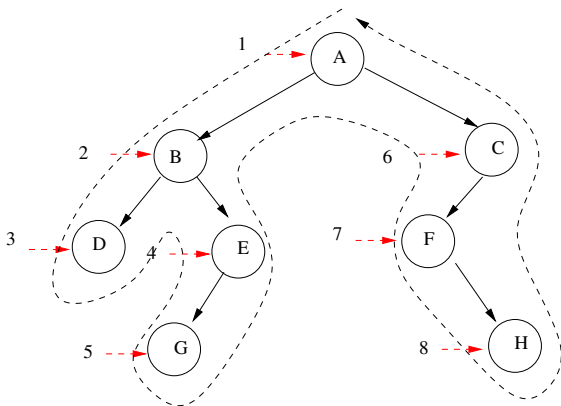
No percurso em pré-ordem

- visita-se a raiz, depois
- percorre-se a subárvore esquerda em pré-ordem, e por fim
- percorre-se a subárvore direita em pré-ordem.

Percurso em pré-ordem

No percurso em pré-ordem

- visita-se a raiz, depois
- percorre-se a subárvore esquerda em pré-ordem, e por fim
- percorre-se a subárvore direita em pré-ordem.



A, B, D, E, G, C, F, H

Percurso em ordem simétrica

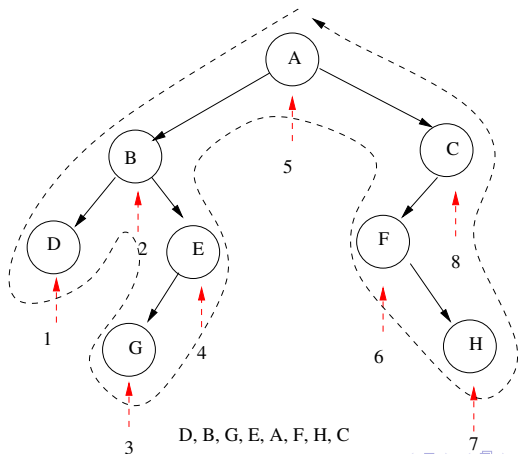
No percurso em ordem simétrica

- percorre-se a subárvore esquerda em ordem simétrica, depois
- visita-se a raiz, e por fim
- percorre-se a subárvore direita em ordem simétrica.

Percurso em ordem simétrica

No percurso em ordem simétrica

- percorre-se a subárvore esquerda em ordem simétrica, depois
- visita-se a raiz, e por fim
- percorre-se a subárvore direita em ordem simétrica.



Percurso em pós-ordem

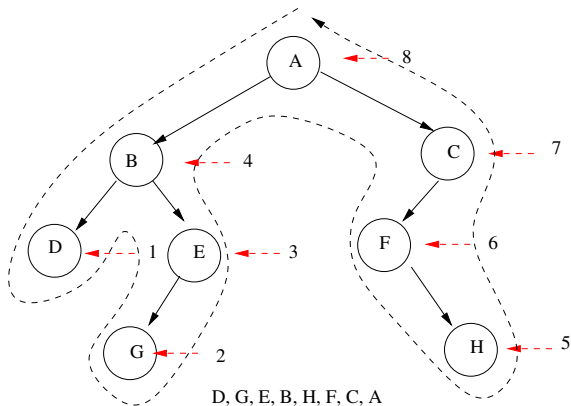
No percurso em pós-ordem

- percorre-se a subárvore esquerda em pós-ordem, depois
- percorre-se a subárvore direita em pós-ordem, e por fim
- visita-se a raiz.

Percurso em pós-ordem

No percurso em pós-ordem

- percorre-se a subárvore esquerda em pós-ordem, depois
- percorre-se a subárvore direita em pós-ordem, e por fim
- visita-se a raiz.



Construção da árvore binária

Não conseguimos construir a árvore dado um único percurso, mas conseguimos construí-la a partir de

Construção da árvore binária

Não conseguimos construir a árvore dado um único percurso, mas conseguimos construí-la a partir de

- dois percursos em profundidade distintos,

Construção da árvore binária

Não conseguimos construir a árvore dado um único percurso, mas conseguimos construí-la a partir de

- dois percursos em profundidade distintos,
- um percurso e uma informação binária que indique para cada nó a existência ou não dos filhos à esquerda e à direita (e.g., em pré-ordem $A_{11}B_{11}D_{00}E_{10}G_{00}C_{10}F_{01}H_{00}$), e

Construção da árvore binária

Não conseguimos construir a árvore dado um único percurso, mas conseguimos construí-la a partir de

- dois percursos em profundidade distintos,
- um percurso e uma informação binária que indique para cada nó a existência ou não dos filhos à esquerda e à direita (e.g., em pré-ordem $A_{11}B_{11}D_{00}E_{10}G_{00}C_{10}F_{01}H_{00}$), e
- um percurso com notação parentética, que indica subárvore esquerda, raiz, e subárvore direita para cada nó. Por exemplo, em ordem simétrica
 $((()D())B((()G())E()))A((()F((()H()))C()))$.

Construção da árvore binária

Suponha, por exemplo, os percursos abaixo:

- ordem simétrica D, B, G, E, A, F, H, C, e
- pós-ordem D, G, E, B, H, F, C, A.

Construção da árvore binária

Suponha, por exemplo, os percursos abaixo:

- ordem simétrica D, B, G, E, A, F, H, C, e
- pós-ordem D, G, E, B, H, F, C, A.
- Sabemos que o último nó da pós-ordem é a raiz da árvore
- e que na ordem simétrica ele se encontra entre as subárvores esquerda e direita.

Construção da árvore binária

Suponha, por exemplo, os percursos abaixo:

- ordem simétrica D, B, G, E, A, F, H, C, e
- pós-ordem D, G, E, B, H, F, C, A.
- Sabemos que o último nó da pós-ordem é a raiz da árvore
- e que na ordem simétrica ele se encontra entre as subárvores esquerda e direita.

Isto sugere o seguinte algoritmo recursivo para construção de uma árvore binária a partir dos percursos em ordem simétrica e pós-ordem.

Construção da árvore binária

- 1 Se o percurso tiver um único nó, então retorna o apontador da árvore de grau zero, altura um e informação dada. Caso contrário,
- 2 Obtém a raiz como último elemento do percurso em pós-ordem e calcula os números de nós das subárvores esquerda e direita no percurso em ordem simétrica.
- 3 Separa os percursos em ordem simétrica e pós-ordem das subárvores esquerda e direita.
- 4 Constrói as subárvores esquerda e direita retornando os apontadores de suas raízes.
- 5 Cria o nó raiz com as respectivas subárvores esquerda e direita, e retorna seu apontador.