CSE 203: Trees

Dr. Mohammed Eunus Ali
Professor
CSE, BUET

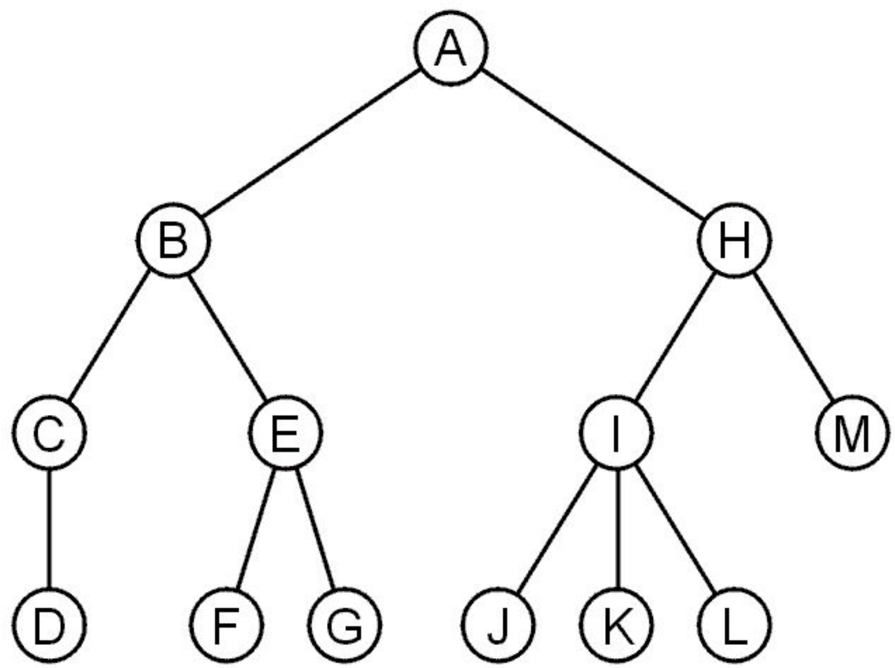
Outline

- Definition of a tree data structure and its components
- Concepts of:
  - Root, internal, and leaf nodes
  - Parents, children, and siblings
  - Paths, path length, height, and depth
  - Ancestors and descendants
  - Ordered and unordered trees
  - Subtrees
- Examples
  - XHTML and CSS

Trees

A rooted tree data structure stores information in *nodes*

- Similar to linked lists:
  - There is a first node, or *root*
  - Each node has variable number of references to successors
  - Each node, other than the root, has exactly one node pointing to it



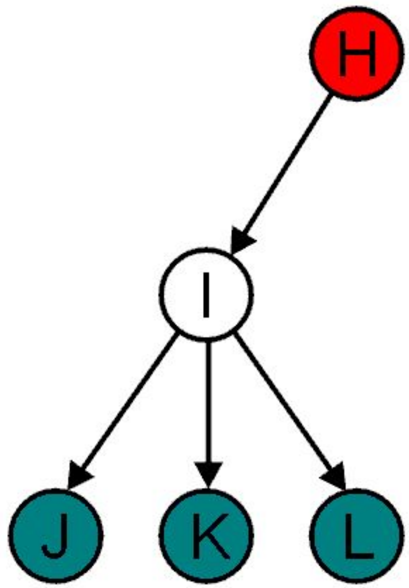
Terminology

All nodes will have zero or more child nodes or *children*

- I has three children: J, K and L

For all nodes other than the root node, there is one parent node

- H is the parent I

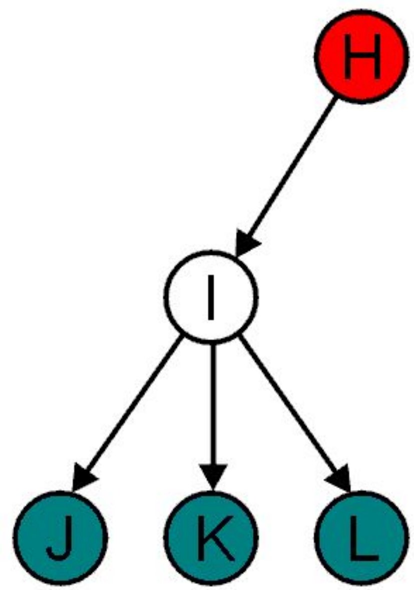


Terminology

The *degree* of a node is defined as the number of its children: $\deg(\mathrm{I}) = 3$

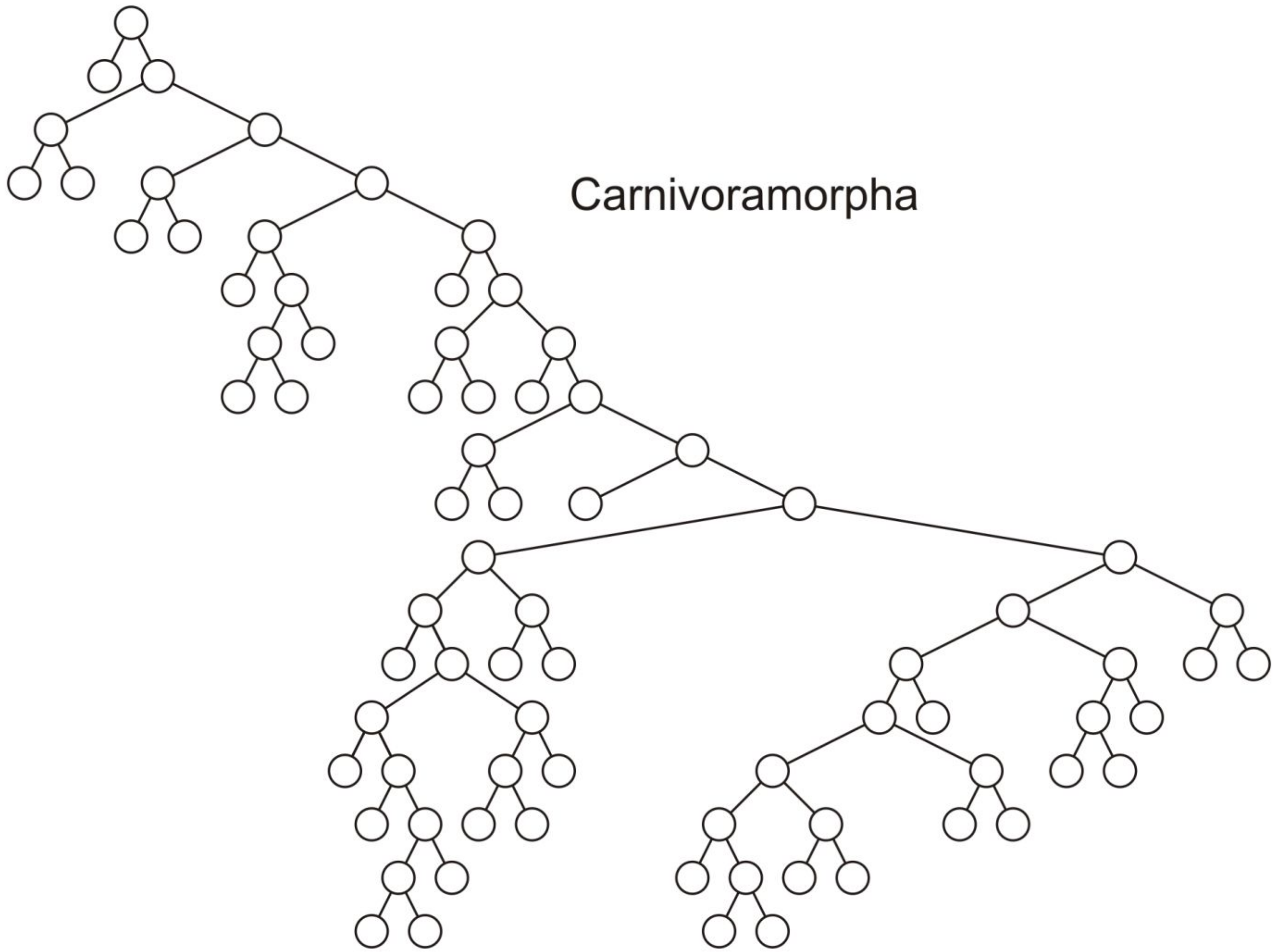
Nodes with the same parent are *siblings*

- J, K, and L are siblings



Terminology

Phylogenetic trees have nodes with degree 2 or 0:

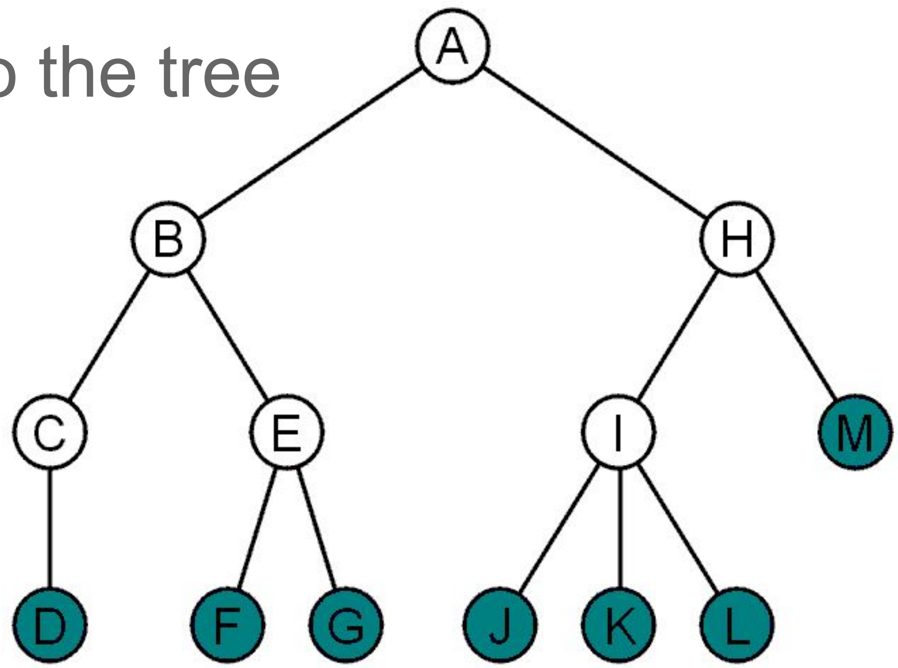


Wesley-Hunt, G. D.; Flynn, J. J. "Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of 'Miacoidea'

Terminology

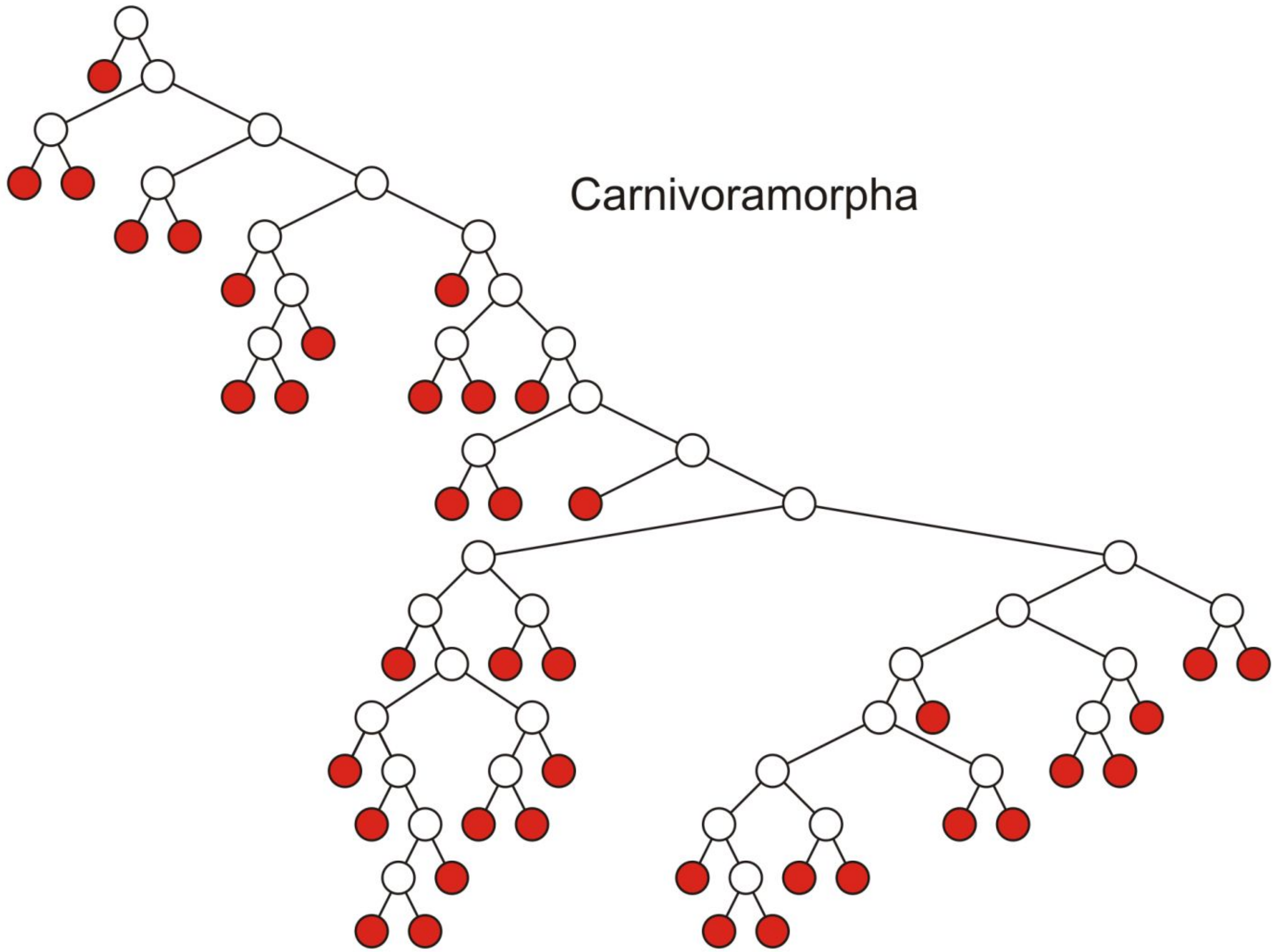
Nodes with degree zero are also called *leaf nodes*

All other nodes are said to be *internal nodes*, that is, they are internal to the tree



Terminology

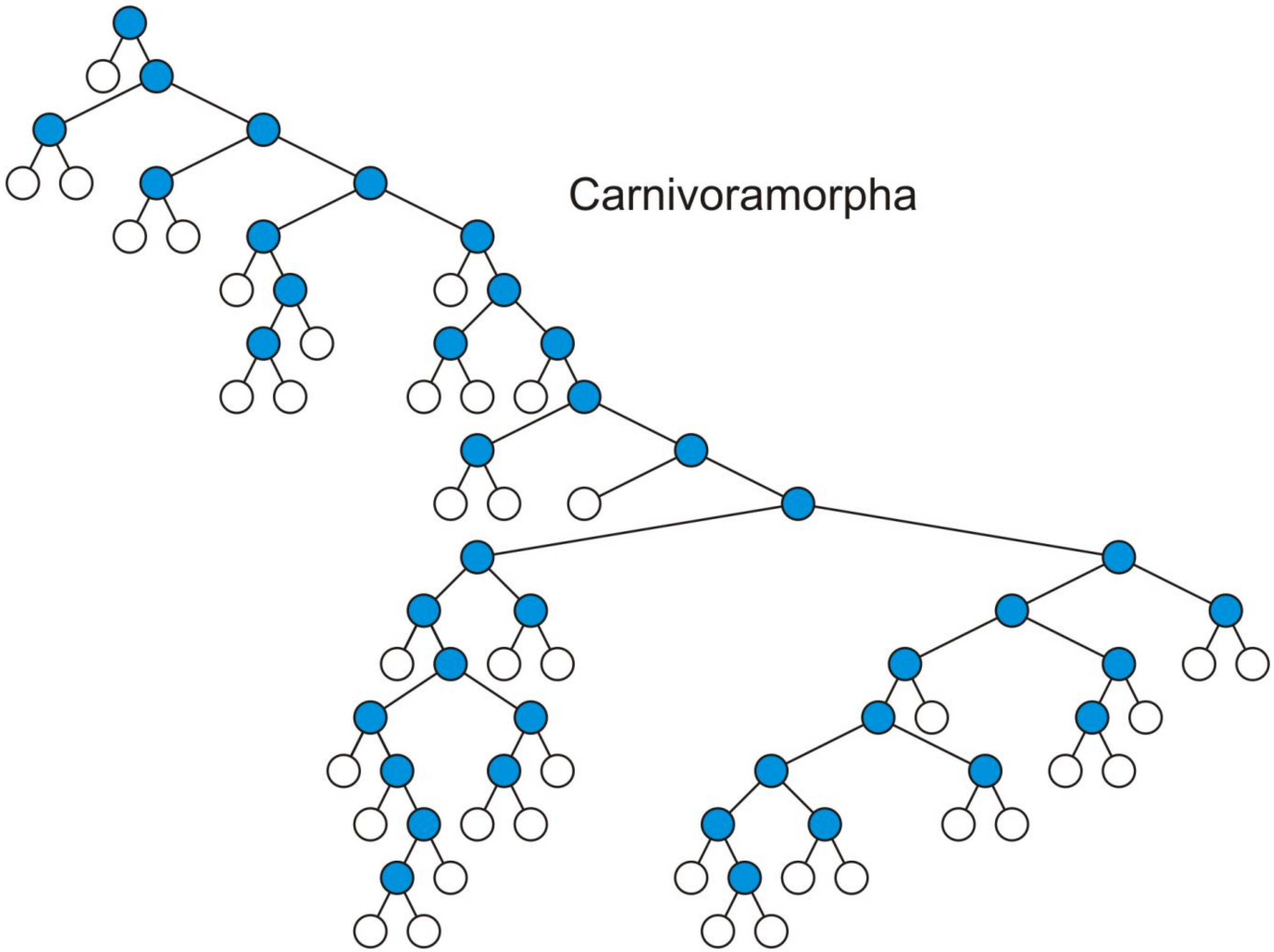
Leaf nodes:



Wesley-Hunt, G. D.; Flynn, J. J. "Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of 'Miacoidea'

Terminology

Internal nodes:

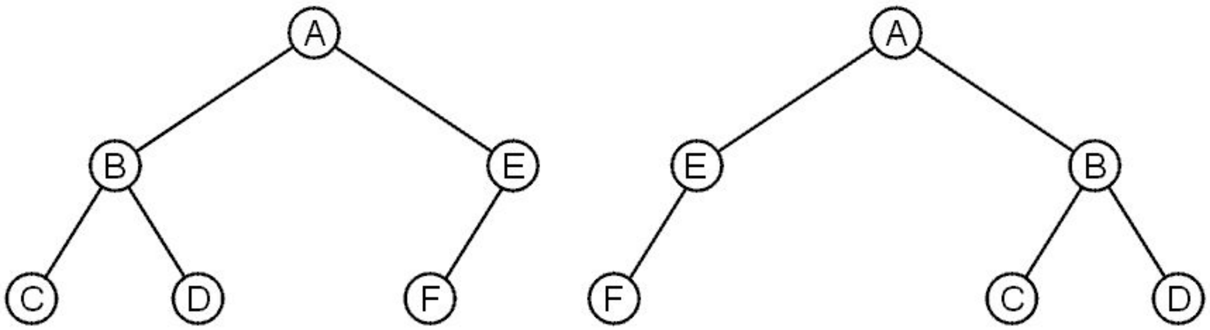


Wesley-Hunt, G. D.; Flynn, J. J. "Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of 'Miacoidea'

Terminology

These trees are equal if the order of the children is ignored

- *unordered trees*

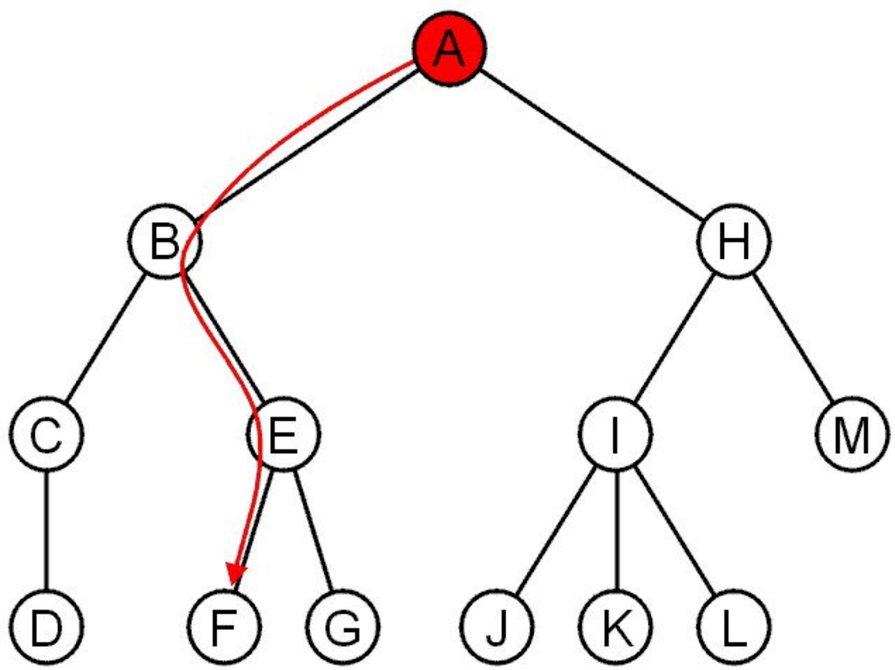


They are different if order is relevant (*ordered trees*)

- We will usually examine ordered trees (linear orders)
- In a hierarchical ordering, order is not relevant

Terminology

The shape of a rooted tree gives a natural flow from the *root node*, or just *root*



Terminology

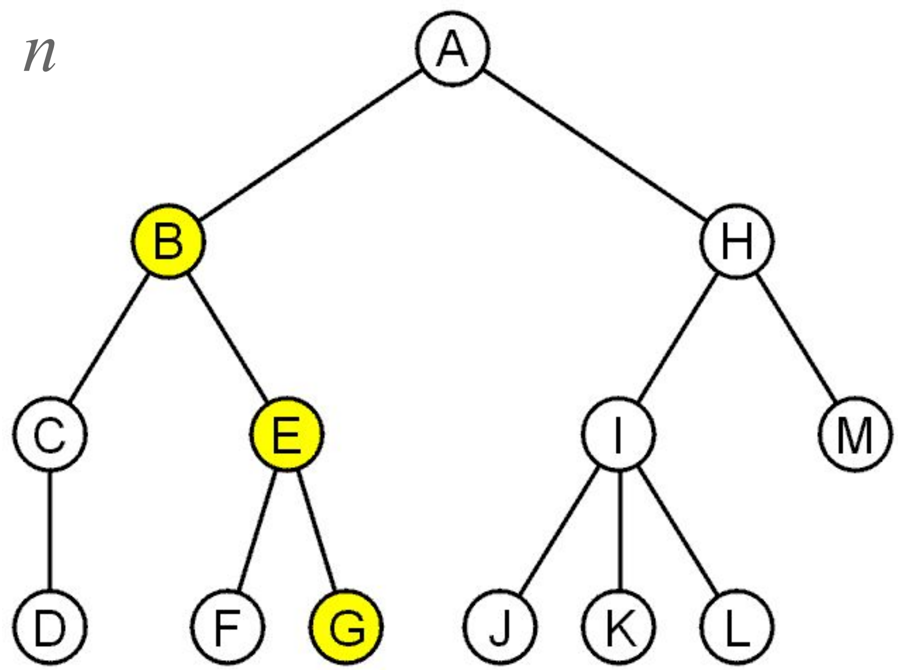
A path is a sequence of nodes

$$(a_0, a_1, \ldots, a_n)$$

where a_{k+1} is a child of a_k is

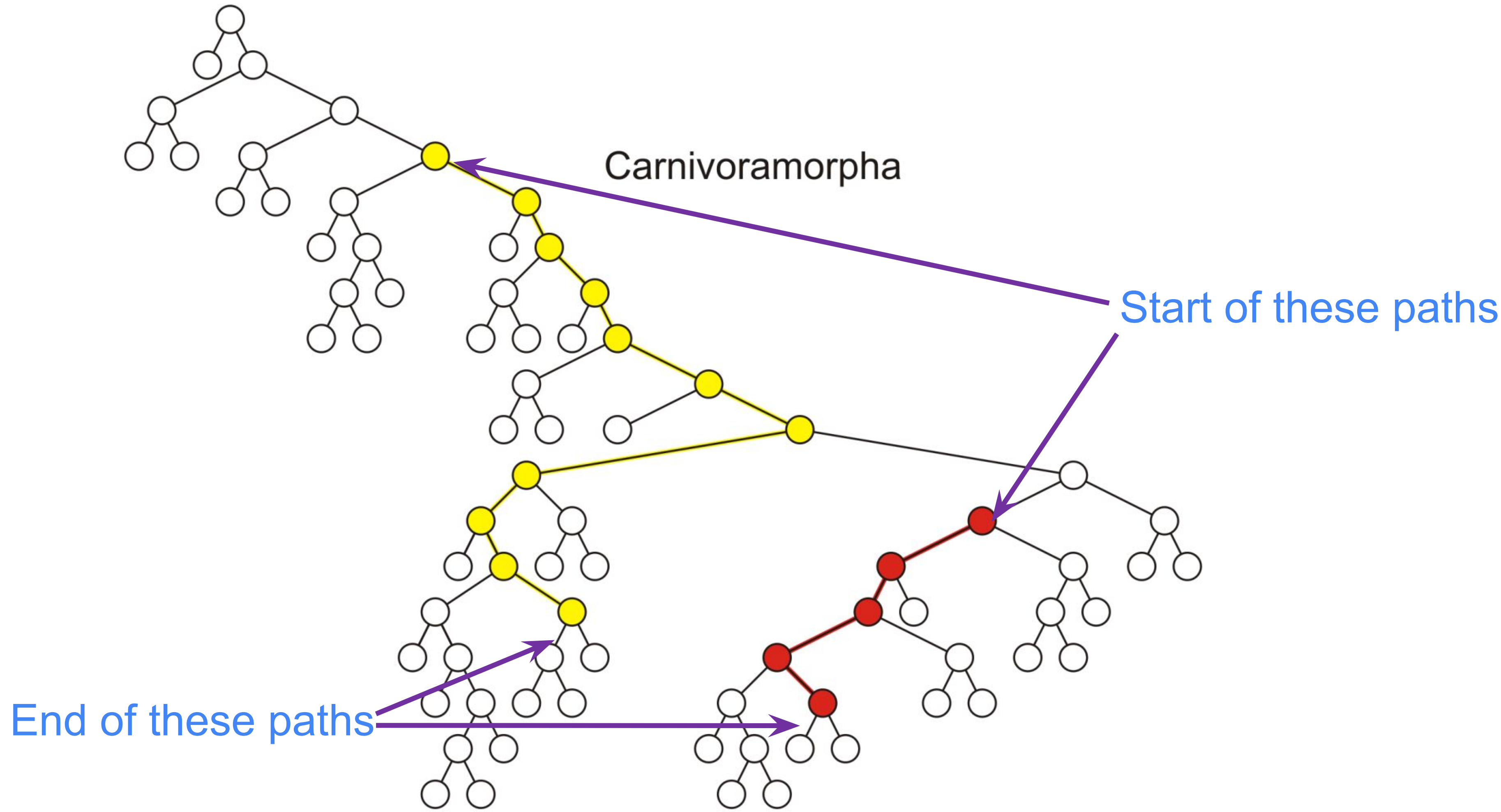
The length of this path is n

E.g., the path (B, E, G) has length 2



Terminology

Paths of length 10 (11 nodes) and 4 (5 nodes)



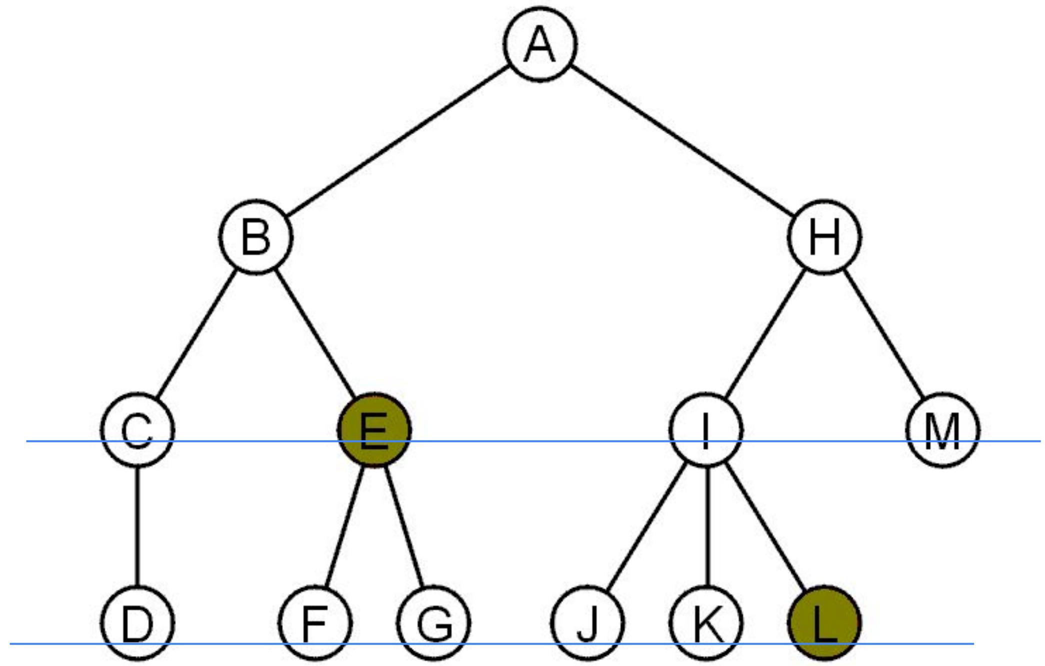
Wesley-Hunt, G. D.; Flynn, J. J. “Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of ‘Miacoidea’

Terminology

For each node in a tree, there exists a unique path from the root node to that node

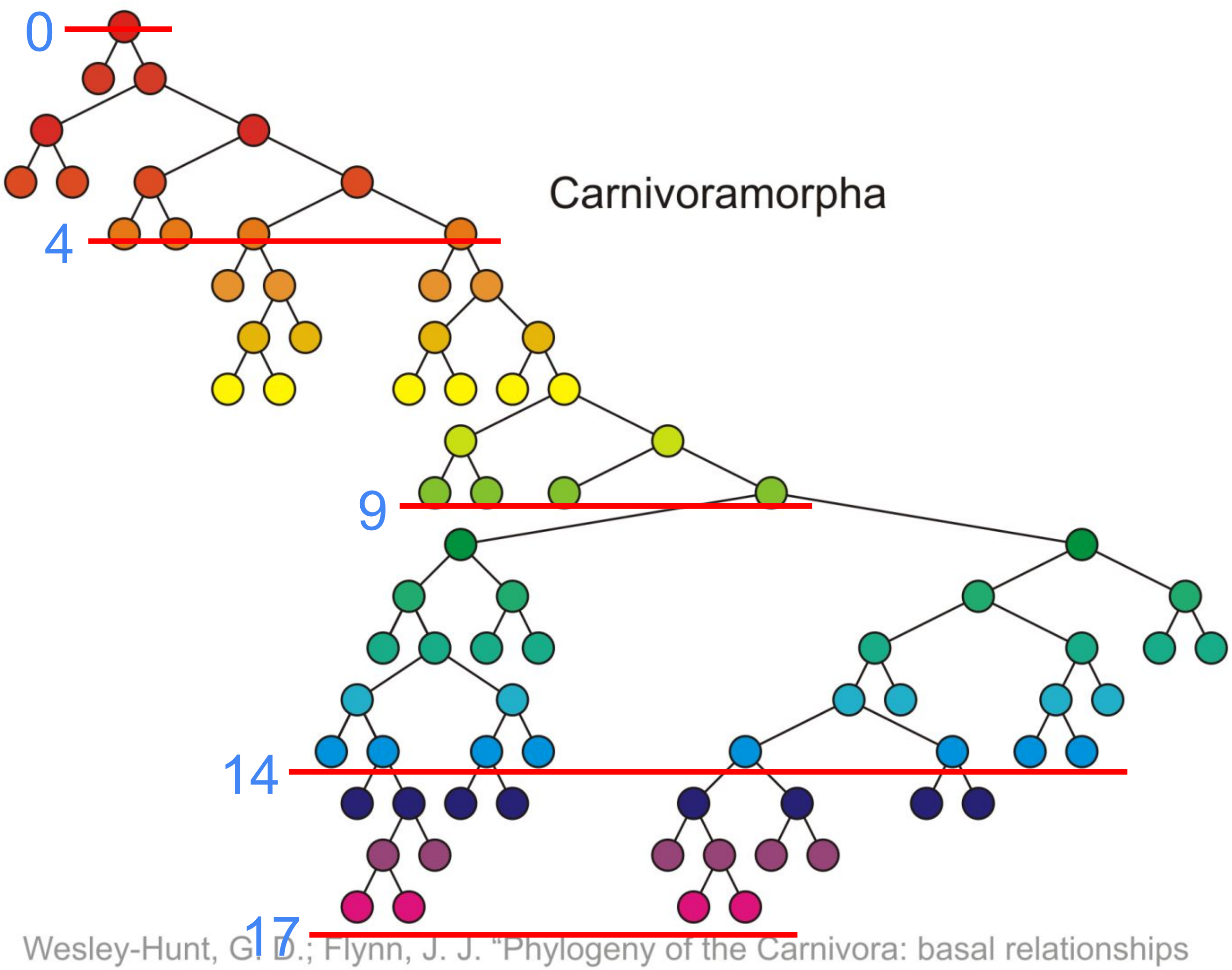
The length of this path is the *depth* of the node, *e.g.*,

- E has depth 2
- L has depth 3



Terminology

Nodes of depth up to 17



Wesley-Hunt, G. D.; Flynn, J. J. "Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of 'Miacoidea'

Terminology

The *height* of a tree is defined as the maximum depth of any node within the tree

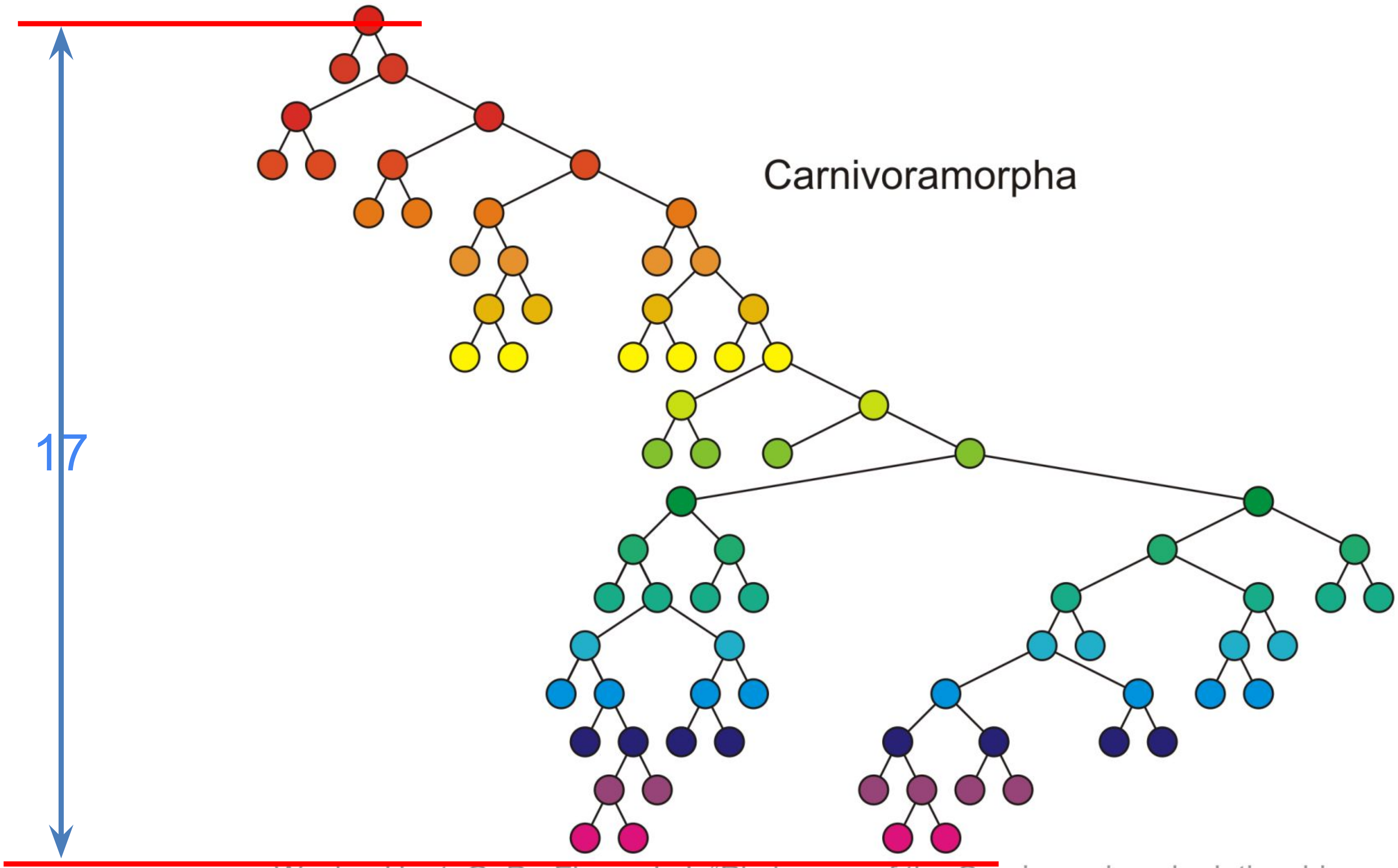
The height of a tree with one node is 0

- Just the root node

For convenience, we define the height of the empty tree to be -1

Terminology

The height of this tree is 17



Wesley-Hunt, G. D.; Flynn, J. J. "Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of 'Miacoidea'

Terminology

If a path exists from node a to node b:

- a is an *ancestor* of b
- b is a *descendent* of a

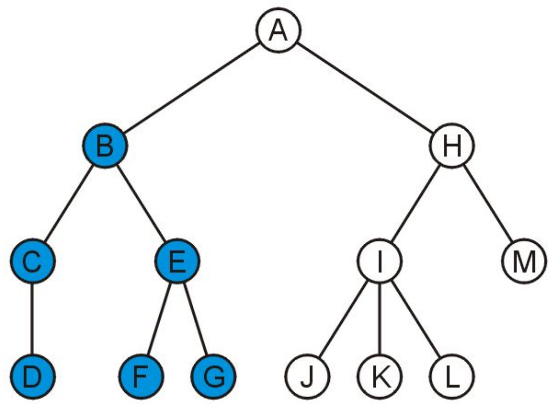
Thus, a node is both an ancestor and a descendant of itself

- We can add the adjective *strict* to exclude equality: a is a *strict descendent* of b if a is a descendant of b but $a \neq b$

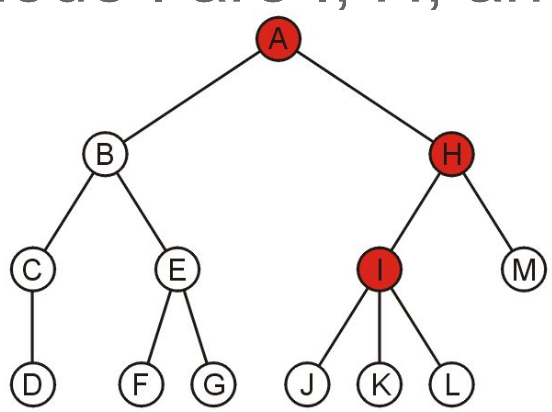
The root node is an ancestor of all nodes

Terminology

The descendants of node B are B, C, D, E, F, and G:

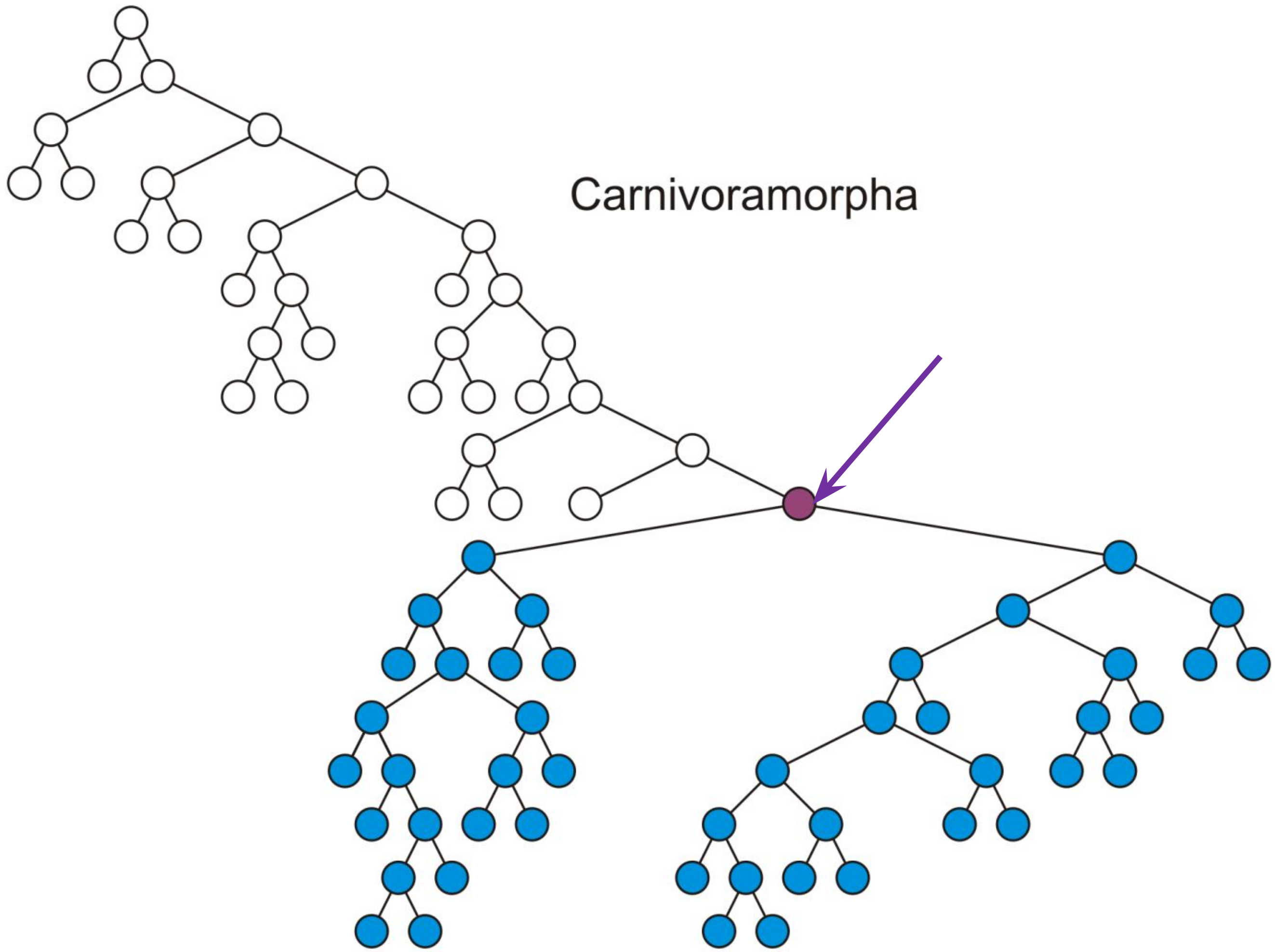


The ancestors of node I are I, H, and A:



Terminology

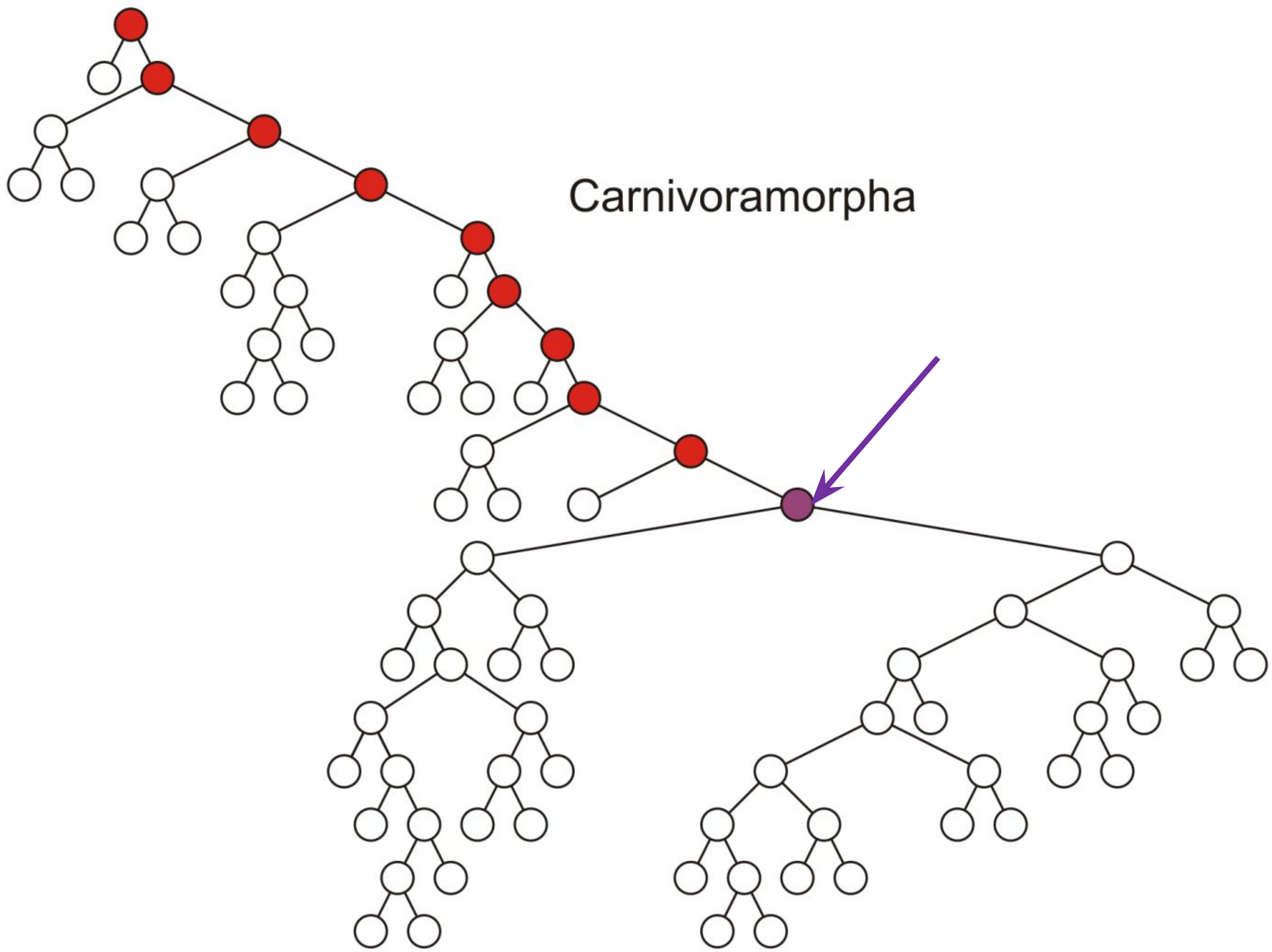
All descendants (including itself) of the indicated node



Wesley-Hunt, G. D.; Flynn, J. J. “Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of ‘Miacoidea’

Terminology

All ancestors (including itself) of the indicated node



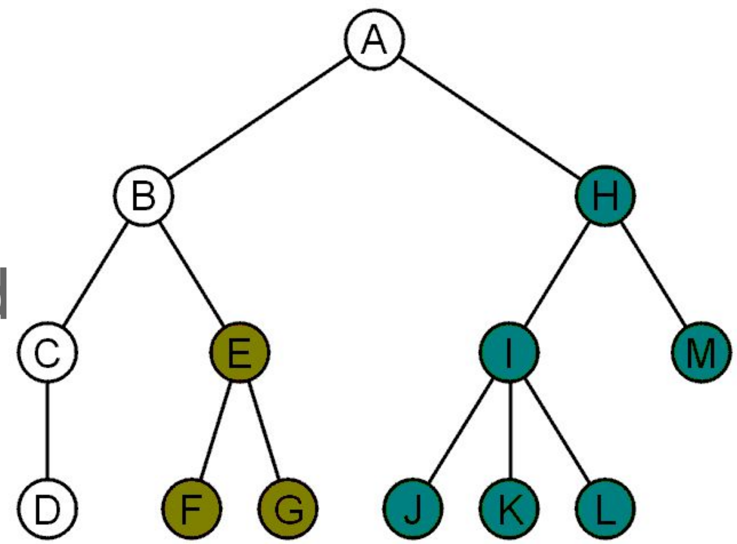
Wesley-Hunt, G. D.; Flynn, J. J. “Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of ‘Miacoidea’

Terminology

Another approach to a tree is to define the tree recursively:

- A degree-0 node is a tree
- A node with degree n is a tree if it has n children and all of its children are disjoint trees (*i.e.*, with no intersecting nodes)

Given any node a within a tree with root r, the collection of a and all of its descendants is said to be a *subtree of the tree with root* a



Example: XHTML and CSS

The XML of XHTML has a tree structure

Cascading Style Sheets (CSS) use the tree structure to modify the display of HTML

Example: XHTML and CSS

Consider the following XHTML document

```
<html>
    <head>
        <title>Hello World!</title>
    </head>
    <body>
        <h1>This is a <u>Heading</u></h1>

        <p>This is a paragraph with some
        <u>underlined</u> text.</p>
    </body>
</html>
```

4.1.3

Example: XHTML and CSS

Consider the following XHTML document

```
<html>
    <head>
        <title>Hello World!</title>
    </head>
    <body>
        <h1>This is a <u>Heading</u></h1>

        <p>This is a paragraph with some
        <u>underlined</u> text.</p>
    </body>
</html>
```

title

heading

body of page

underlining

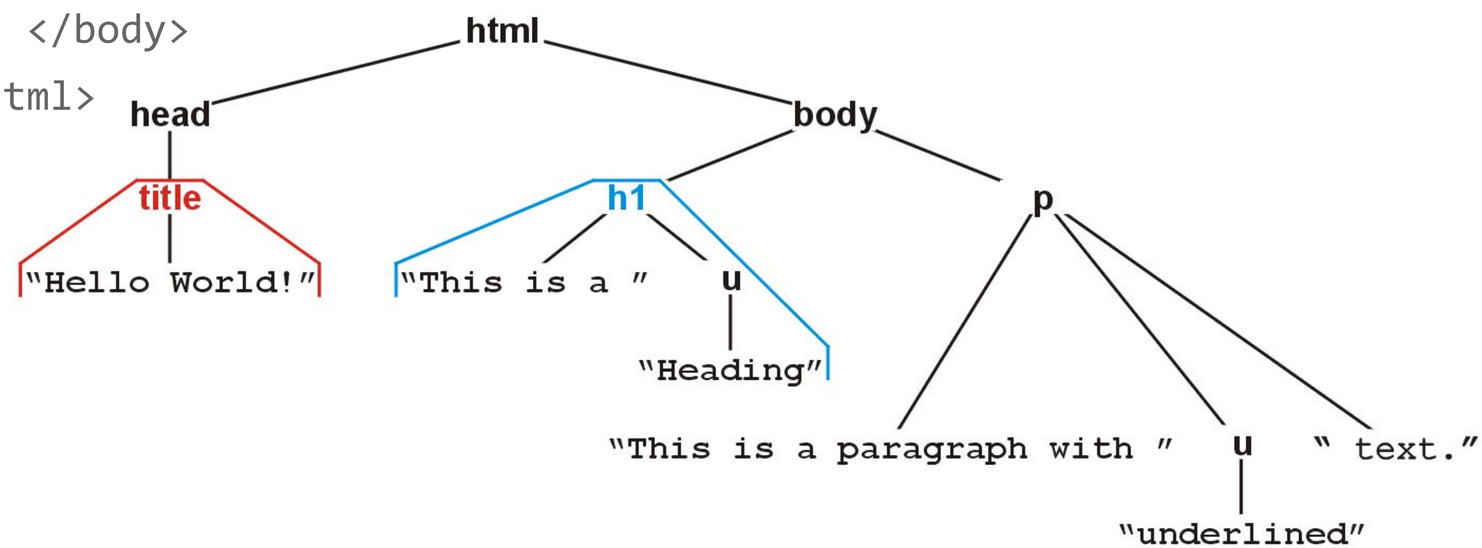
paragraph

Example: XHTML and CSS

The nested tags define a tree rooted at the HTML tag

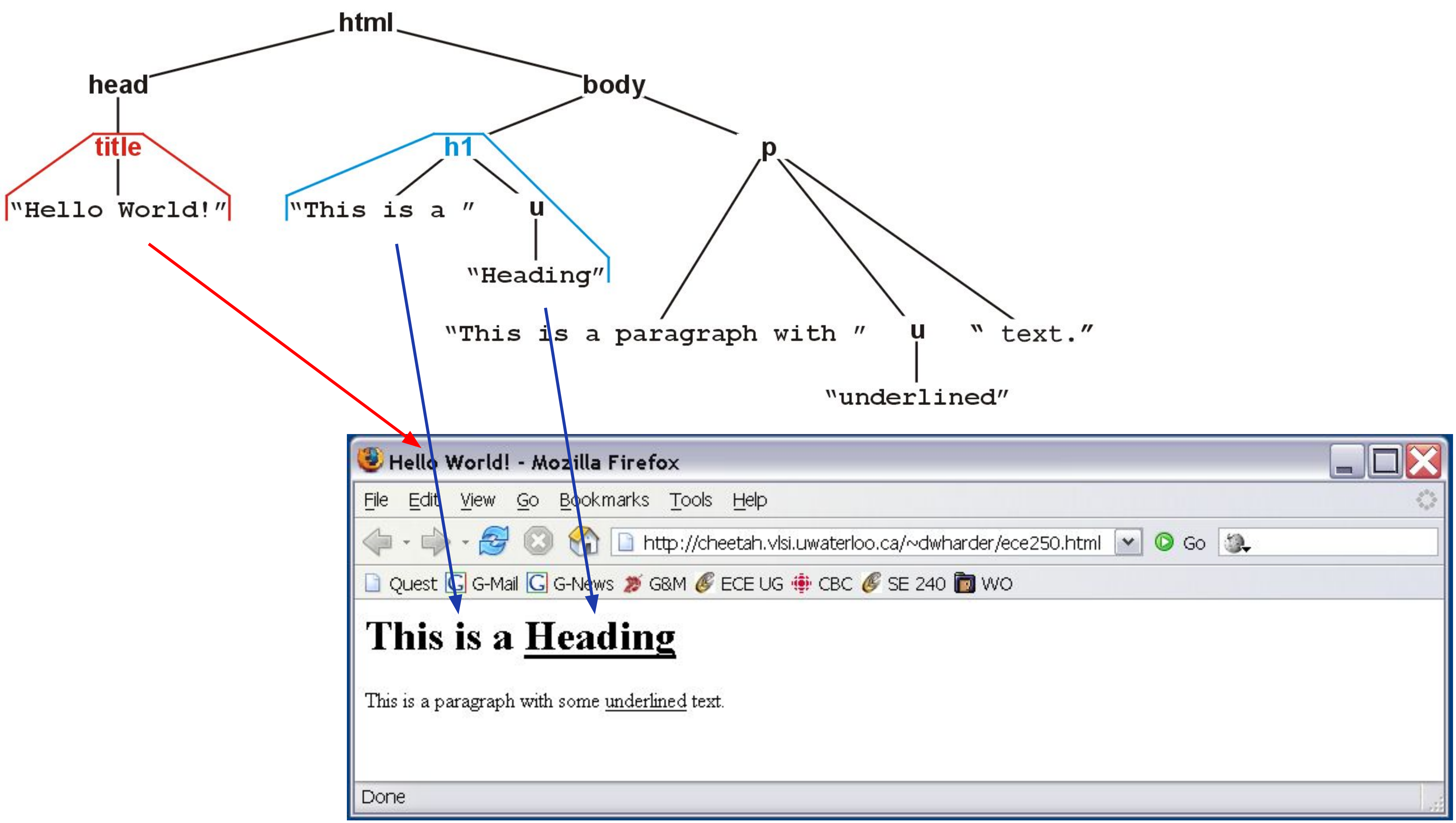
```html
<html>
    <head>
        <title>Hello World!</title>
    </head>
    <body>
        <h1>This is a <u>Heading</u></h1>

        <p>This is a paragraph with some
        <u>underlined</u> text.</p>
    </body>
</html>
```



Example: XHTML and CSS

Web browsers render this tree as a web page



Example: XHTML and CSS

Cascading Style Sheets (CSS) make use of this tree structure to describe how HTML should be displayed

- For example:

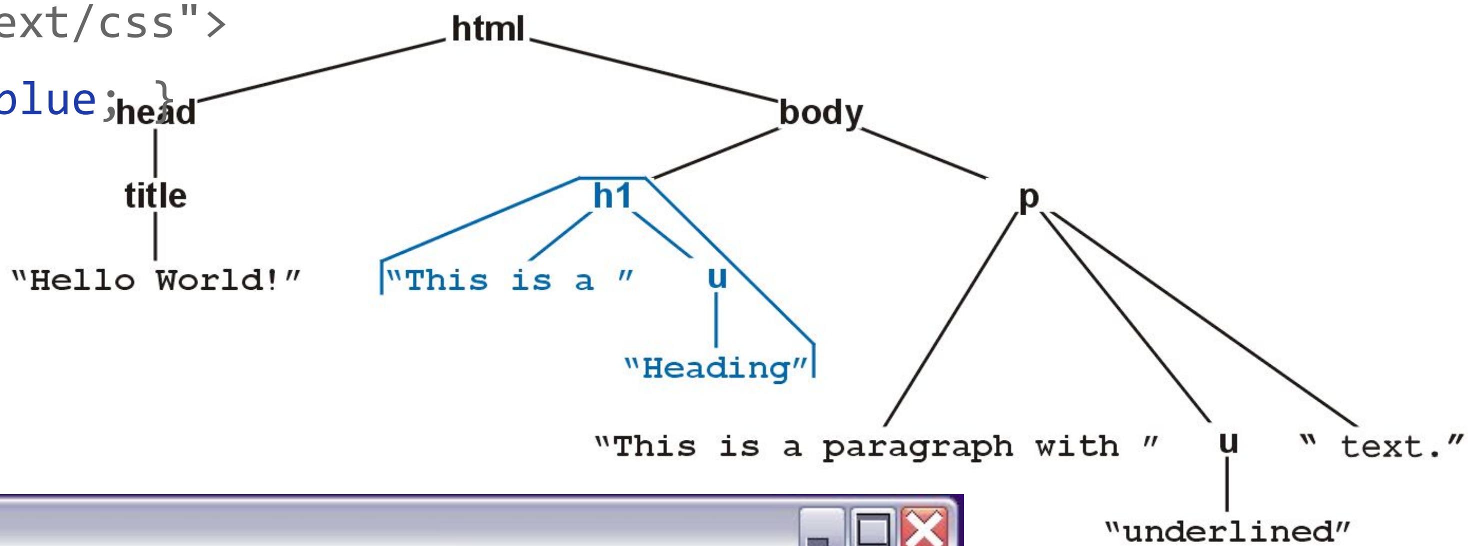
```
<style type="text/css">
    h1 { color:blue; }
</style>
```

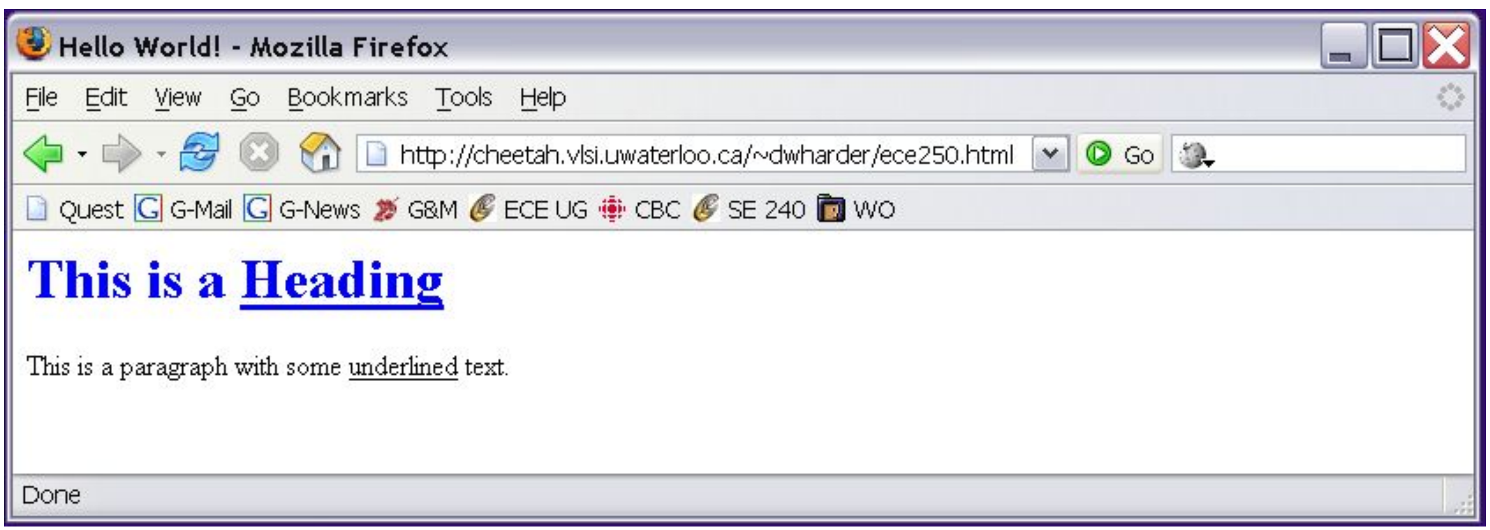
  indicates all text/decorations descendant from an `h1` header should be blue

Example: XHTML and CSS

For example, this style renders as follows:

```
<style type="text/css">
   h1 { color:blue; }
</style>
```

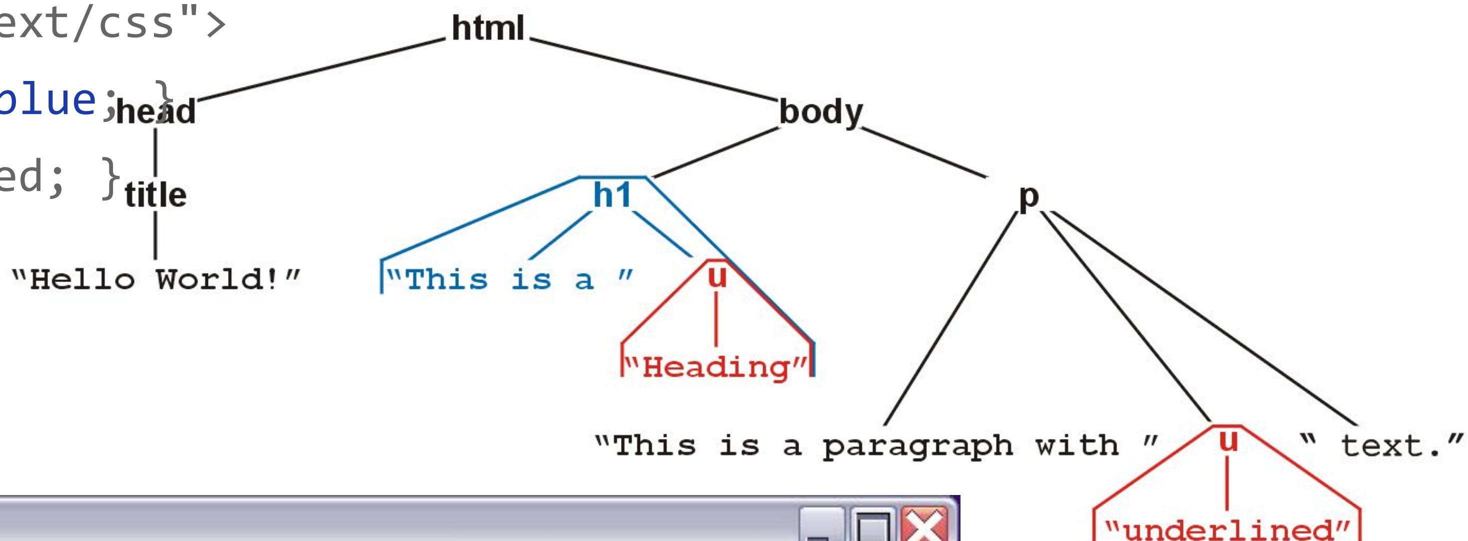




Example: XHTML and CSS

For example, this style renders as follows:

```
<style type="text/css">
    h1 { color:blue; }
    u { color:red; }
</style>
```



Example: XHTML and CSS

Suppose you don't want underlined items in headers (`h1`) to be red

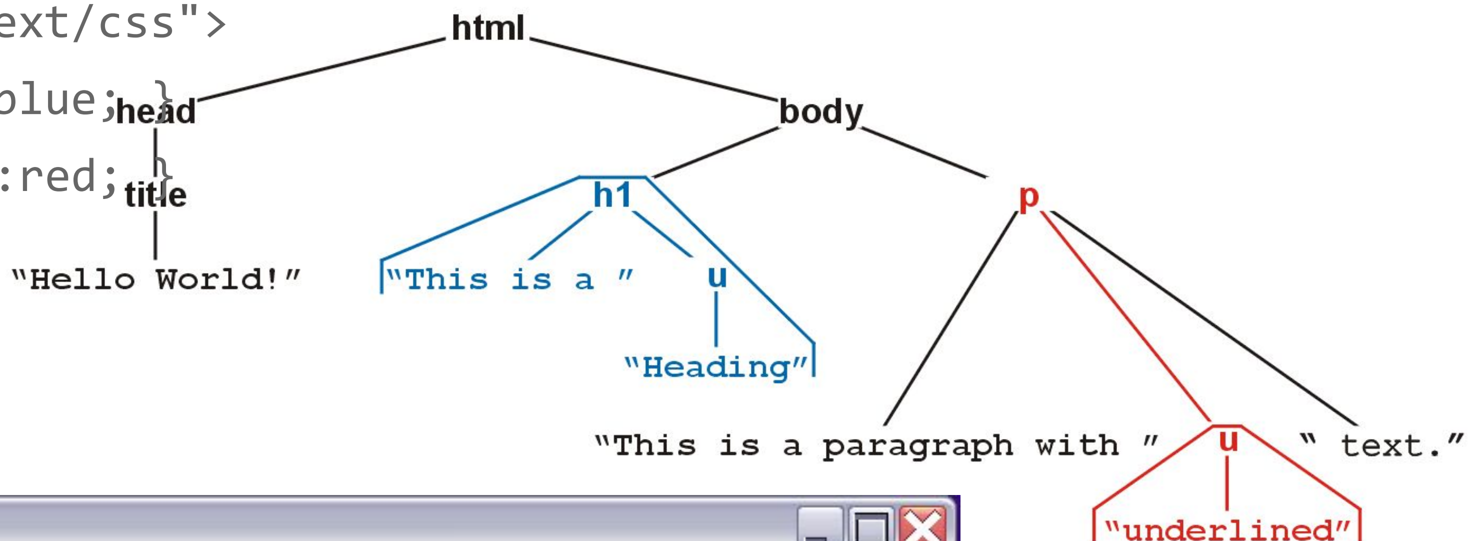
- More specifically, suppose you want any underlined text within paragraphs to be red

That is, you only want text marked as `<u>text</u>` to be underlined if it is a descendant of a `<p>` tag

Example: XHTML and CSS

For example, this style renders as follows:

```
<style type="text/css">
   h1 { color:blue; }
   p u { color:red; }
</style>
```



Example: XHTML and CSS

You can read the second style

```
<style type="text/css">
   h1 { color:blue; }
   p u { color:red; }
</style>
```

as saying “text/decorations descendant from the underlining tag (<u>) which itself is a descendant of a paragraph tag should be coloured red”

Summary

In this topic, we have:

- Introduced the terminology used for the tree data structure
- Discussed various terms which may be used to describe the properties of a tree, including:
  - root node, leaf node
  - parent node, children, and siblings
  - ordered trees
  - paths, depth, and height
  - ancestors, descendants, and subtrees
- We looked at XHTML and CSS

References

[1] Donald E. Knuth, *The Art of Computer Programming, Volume 1: Fundamental Algorithms*, 3rd Ed., Addison Wesley, 1997, §2.2.1, p.238.