CS 225

#13: Trees

February 14, 2018 · *Wade Fagen-Ulmschneider*

Iterator Design:

[Monday's Lecture]: To implement an iterator, the implementing class must have two member functions:

- ::begin(), returns an iterator at the first element
- ::end(), returns an iterator one past the end

Queue.h

```
4  template <class QE>
5  class Queue {
6    public:
7      class QueueIterator :
       public std::iterator<std::forward_iterator_tag, QE> {
8        public:
9          QueueIterator(unsigned index);
10         QueueIterator& operator++();
11         bool operator==(const QueueIterator &other);
12         bool operator!=(const QueueIterator &other);
13         QE& operator*();
14         QE* operator->();
15       private:
16         int location_;
17
18     };
19
20
21
22   /* ... */
23   private:
24     QE* arr_; unsigned capacity_, count_, entry_, exit_;
25 };
```

How does the `Queue` and the `QueueIterator` interact?

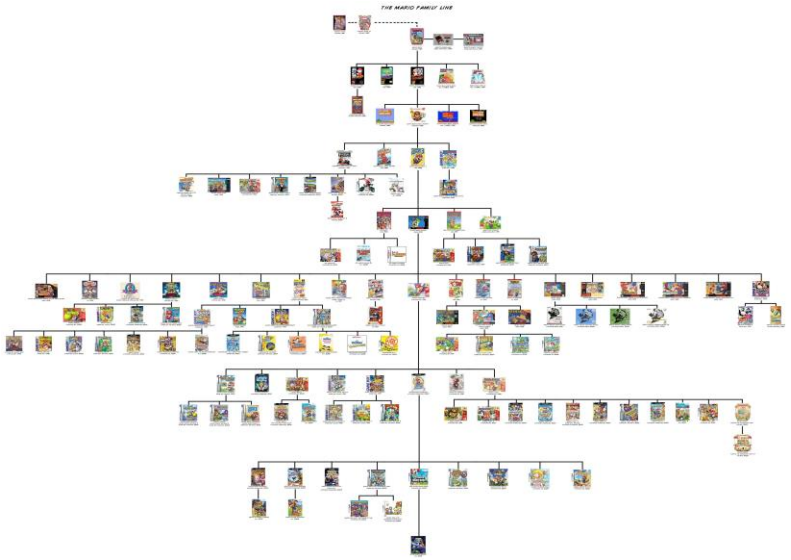
Two big takeaways:

1.

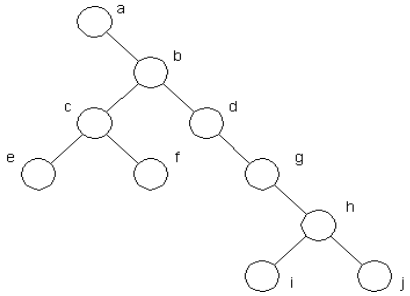
2.

Trees!

"The most important non-linear data structure in computer science."
- David Knuth, The Art of Programming, Vol. 1



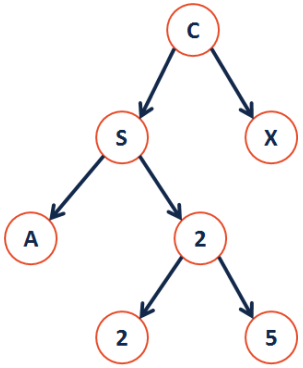
- We will primarily talk about **binary trees**
- What's the longest "word" you can make using the **vertex** labels in the tree (repeats allowed)?
- Find an **edge** that is not on the longest **path** in the tree. Give that edge a reasonable name.
- One of the vertices is called the **root** of the tree. Which one?



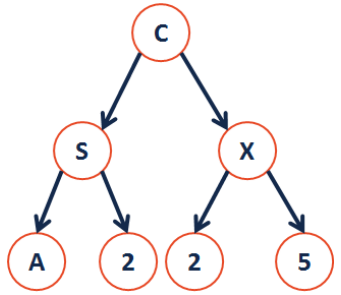
- Make a "word" containing the names of the vertices that have a **parent** but no **sibling**.
- How many parents does each vertex have?
- Which vertex has the fewest **children**?
- Which vertex has the most **ancestors**?
- Which vertex has the most **descendants**?
- List all the vertices is b's left **subtree**.
- List all the **leaves** in the tree.

Definition: Binary Tree

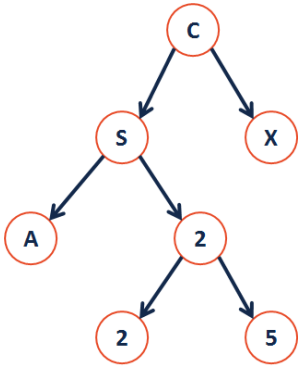
A *binary tree* **T** is either:



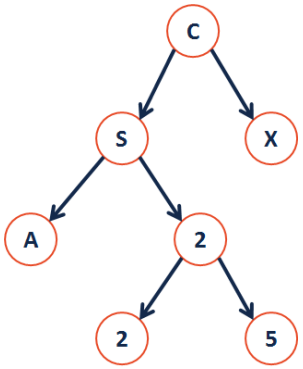
Tree Property: Perfect



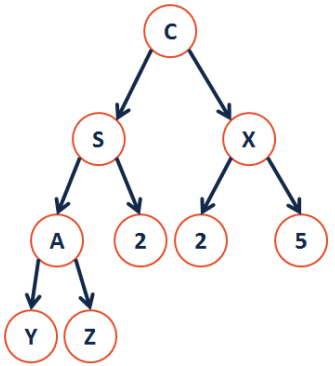
Tree Property: Tree Height



Tree Property: Full



Tree Property: Complete



CS 225 – Things To Be Doing:

1. Programming Exam A is ongoing
2. MP3 has been released; extra credit deadline is Monday!
3. lab_quacks in lab this week
4. Daily POTDs