Homework 4

* Due the 20th, along with sets lab and priorty queue lab.
* Should not past O(n).

Tree Sort

Page 347 code

* Study this it will probably be on the final

Priority Queue (Heap)

* It is a complete binary tree
* Everything is pushed up and to the left.
* Every node is greater then or equal to its children. Putting the largest element on top of the tree. Or the first element in the vector/array.
* Find the nodes children
  1. Node I - left child is 2i + 1 – right child is 2i + 2
  2. Node I – parent is (I-1)/2.

Priority Quee Test program

* #include <queue>
* Priority\_queue<int> pq;
* Pq.push(2);
* Pq.push(4);
* Pq.push(3);
* Pq.push(1);
* Cout << pq.top(); -> 4 has to put the largest value at the top.
* Pq.pop(); pops the root element – removes 4
* Cout << pq.top() - > 3;

Priority Queue Book Implementation

* Unlike the book we want to use a template class T and use a vectory as the private data.
* All we will have to write is the push and pop for the lab. Everything else uses the pre existing vector methods. IE Front(), Size(), Empty().
* Push Heap
  + Void push\_heap(iterator start, iterator stop)
    - Unsigned int position = (stop – start) - 1;
    - Unsigned int parent = (position –1) /2;
    - While (position > 0 && start[position] > start[parent])
      * Swap(start[position], start[parent]);
      * Position = parent;
      * Parent = (position –1) /2;
* Pop Heap
  + Void pop\_heap (iterator start, iterator stop)
    - Unsigned int last\_position = (stop – start) -1;
    - Swap(start[0], start[last\_position]);
    - Adjust\_heap(start, last\_position, 0);