

DATA STRUCTURE SELECTION

FINAL WRAP UP

Big-O: Naïve sort

What is the Big-O of the following algorithm?

```
void selectionSort(vector<int>& a, int N){
    //Precondition: unsorted array
    //Post condition: sorted array in ascending order
    for(int i =0; i<N; i++){
        int index=i;
        for(int j = i+1; j<N;j++){
            if(a[j]<a[index]){
                index = j;
            }
        }
        int tmp = a[i];
        a[i] = a[index];
        a[index]=tmp;
    }
}
```

Data structure application

Use one of the data structures we have learned to improve the running time.

```
void selectionSort(vector<int>& a, int N){
    //Precondition: unsorted array
    //Post condition: sorted array in ascending order
    for(int i =0; i<N; i++){
        int index=i;
        for(int j = i+1; j<N;j++){
            if(a[j]<a[index]){
                index = j;
            }
        }
        int tmp = a[i];
        a[i] = a[index];
        a[index]=tmp;
    }
}
```

Review copy constructor

Implement the copy constructor for a BST

Data structure Comparison

	Insert	Search	Min	Max	Delete min	Delete max	Delete (any)
Sorted array							
Unsorted array							
Sorted linked list (assume access to both head and tail)							
Unsorted linked list							
Stack							
Queue							
BST (unbalanced)							
BST (balanced)							
Min Heap							
Max Heap							

Data structure Comparison

	Insert	Search	Min	Max	Delete min	Delete max	Delete (any)
Sorted array	$O(N)$	$O(\log N)$	$O(1)$	$O(1)$	$O(N)$ if ascending order, else $O(1)$	$O(1)$ if ascending, else $O(N)$	$O(\log N)$ to find, $O(N)$ to delete
Unsorted array	$O(1)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$
Sorted linked list (assume access to both head and tail)	$O(N)$	$O(N)$	$O(1)$	$O(1)$	$O(1)$	$O(1)$	$O(N)$ to find, $O(1)$ to delete
Unsorted linked list	$O(1)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$ to find, $O(1)$ to delete
Stack	$O(1)$ - only insert to top	Not supported	Not supported	Not supported	Not supported	Not supported	$O(1)$ - Only the element on top of the stack
Queue	$O(1)$ - only to the rear of the queue	Not supported	Not supported	Not supported	Not supported	Not supported	$O(1)$ - only the element at the front of the queue
BST (unbalanced)	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$
BST (balanced)	$O(\log N)$	$O(\log N)$	$O(\log N)$	$O(\log N)$	$O(\log N)$	$O(\log N)$	$O(\log N)$
Min Heap	$O(\log N)$	Not supported	$O(1)$	Not supported	$O(\log N)$	Not supported	$O(\log N)$
Max Heap	$O(\log N)$	Not supported	Not supported	$O(1)$	Not supported	$O(\log N)$	$O(\log N)$