

## PRACTICE MIDTERM EXAM #2 - SOLUTIONS

### 1. ADTs

```
static void printCharPairFrequencies(const Lexicon& english) {  
    Map<string,int> pairs;  
    for (string word : english) {  
        for (int i = 0; i < word.length() - 1; i++) {  
            pairs[word.substr(i,2)]++;  
        }  
    }  
    for (string pair : pairs) {  
        cout << pair << " " << pairs[pair] << endl;  
    }  
}
```

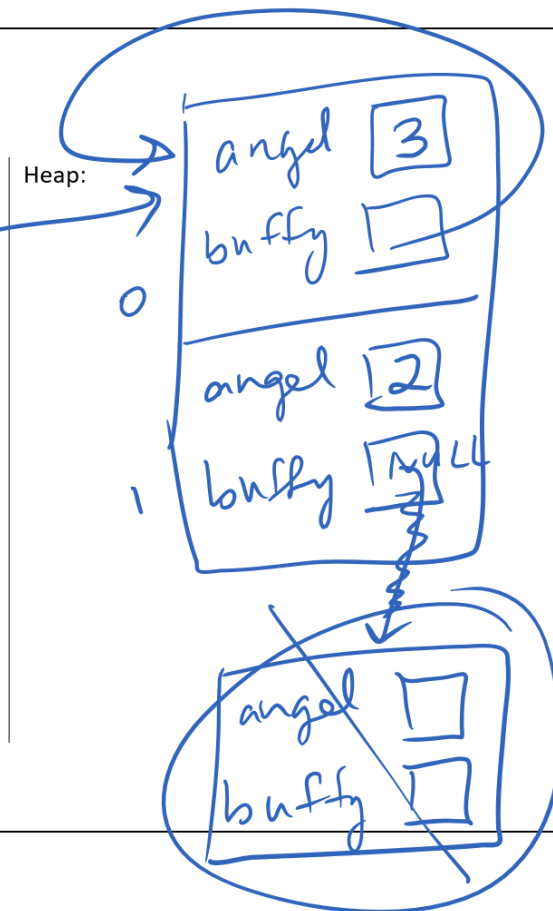
### 2. Pointers and Memory

DRAWING:

Stack:



Heap:



---

## 2. Recursion [by Marty Stepp]

```
string moveToEnd(string s, char c) {
    if (s.length() == 0) {
        return "";
    } else if (toupper(s[0]) == toupper(c)) {
        char upperC = toupper(s[0]);
        return moveToEnd(s.substr(1), c) + upperC;
    } else {
        return s[0] + moveToEnd(s.substr(1), c);
    }
}
```

```
string moveToEnd(string s, char c) {
    if (s == "") {
        return s;
    }
    char first = s.at(0);
    s = s.substr(1, s.length() - 1);
    s = moveToEnd(s, c);
    if (toupper(first) == toupper(c)) {
        char upperFirst = toupper(first);
        return s + upperFirst;
    } else {
        return first + s;
    }
}
```

---

## 3. Linked List Nodes

```
delete list2->next;
list2->next = list->next;
list->next = NULL;
list2->next->next = list;
list = list2;
```

---

## 4. Big-O

O(1)  
O(NlogN)  
O(N<sup>2</sup>)