

CSCI 104L Lecture 3: Recursion

Tail Recursion:

```
int tailRecursion(int input) {  
    if (input == 0) return 0;  
    //do stuff  
    return tailRecursion(input - 1);  
}
```

Examples of Head Recursion:

```
int headRecursion(int input) {  
    if (input == 0) return 0;  
    //do stuff  
    int ret = headRecursion(input - 1);  
    //do more stuff  
    return ret;  
}  
int alsoHeadRecursion(int input) {  
    if (input == 0) return 0;  
    //do stuff  
    return 1+alsoHeadRecursion(input - 1);  
}
```

Indirect Recursion:

```
void indirectRecursion(int input) {  
    if (input == 0) return;  
    otherFunction(input - 1);  
}  
void otherFunction(int input) {  
    indirectRecursion(input);  
}
```

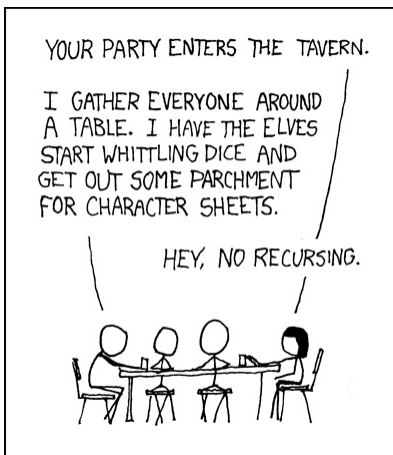


Figure 1: XKCD # 244: Tabletop Roleplaying.

```

//t = target element.  b = array.
//lo = index of first element in array (pass in 0 when you call this function).
//hi = index of last element in array (initially array length-1).
int binarySearch(int t, int *b, int lo, int hi) {
    if (hi < lo) return -1; //nothing to search, it's not in the array.
    else {
        int mid = (hi+lo)/2; //the middle of the array, rounded down.
        if (t == b[mid]) return mid; //found it!
        else if (t < b[mid]) return binarySearch(t, b, lo, mid-1); //search left.
        else return binarySearch(t, b, mid+1, hi); //search right.
    }
}

```

```

//t = target element.  b = array.  len = length of array.
int iterativeBinarySearch(int t, int *b, int len) {
    int lo = 0, hi = len-1, mid;
    while(lo <= hi) {
        mid = (hi+lo)/2;
        if (b[mid]==t) return mid;
        else if (t < b[mid]) hi = mid-1;
        else lo = mid+1;
    }
    return -1;
}

```