

Bryan Melanson

# **How to Learn** **Algorithms**

*When your brain doesn't work*

## Contents

<b>1 Search</b>	<b>2</b>
1.1 Binary Search . . . . .	2

# 1 Search

## 1.1 Binary Search

- Let  $\text{min} = 0$  and  $\text{max} = n-1$ .
- Compute guess as the average of  $\text{max}$  and  $\text{min}$ , rounded down (so that it is an integer).
- If  $\text{array}[\text{guess}]$  equals  $\text{target}$ , then stop. You found it! Return  $\text{guess}$ .
- If the guess was too low, that is,  $\text{array}[\text{guess}] < \text{target}$ , then set  $\text{min} = \text{guess} + 1$ .
- Otherwise, the guess was too high. Set  $\text{max} = \text{guess} - 1$ .
- Go back to step 2.

---

```
// binarysearch.c
int binarySearch(int target){
    int lo = 0; int hi = arrSize;
    while (lo <= hi) {
        int mid = lo + (hi-lo)/2;
        m = getElement(mid);
        if (m == target) return mid;
        if (m > target) hi = mid - 1;
        else lo = mid + 1;
    }
}
```

---