CS 302: Computer Fluency

Paul Bae
pbae@utexas.edu

Announcements

- Homework 2 is being graded
- First Exam is next Monday!!!

Algorithms

A set of specific instructions to accomplish some goal:

Examples:
Sequential Search
Make PB&J
Sum of list

Exercise 1:

Compute and print out the average of a list, if we try to divide by zero, print out an error message

Modules

Modules (or functions) are basically mini-programs that are used for a variety of things

Why use modules?

- Easier to write
- Shorter, cleaner code
- More general
- Layer of abstraction

Exercise 2

Say you have a module FutureValue that calculates the FutureValue of interest in the number of years specified

Give me an algorithm to print out FutureValue for every 12 years from 2000 to 3000

Complexity

Complexity is a notion of how long an algorithm takes
People normally represent this with Big O

Big O

Big O (normally written just O) indicates the complexity of a function

Essentially Big O is the asymptotically bounded limit as the function reaches infinity

Simply put, Big O represents how slow a function grows as input grows bigger and bigger

Example

What is the Big O of the following:
print “Hi Paul”

Example

What is the Big O of:
SequentialSearch

Example

What is the Big O of the following:

```
i = 0
j = 0
while i < n:
    while j < n:
        j++
    i++
```

Example

What is the Big O of the following:
print “Paul”
print “Is”
print “Tired”

Binary Search

SequentialSearch is O(n)
Can we do better?

Searching Algorithms

Let’s think about how we search for a name in the phonebook

We will probably guess where the name is
If it’s a later name (such as Luong), we’ll go to middle
If it’s an early name (such as Bae), we’ll go to the front

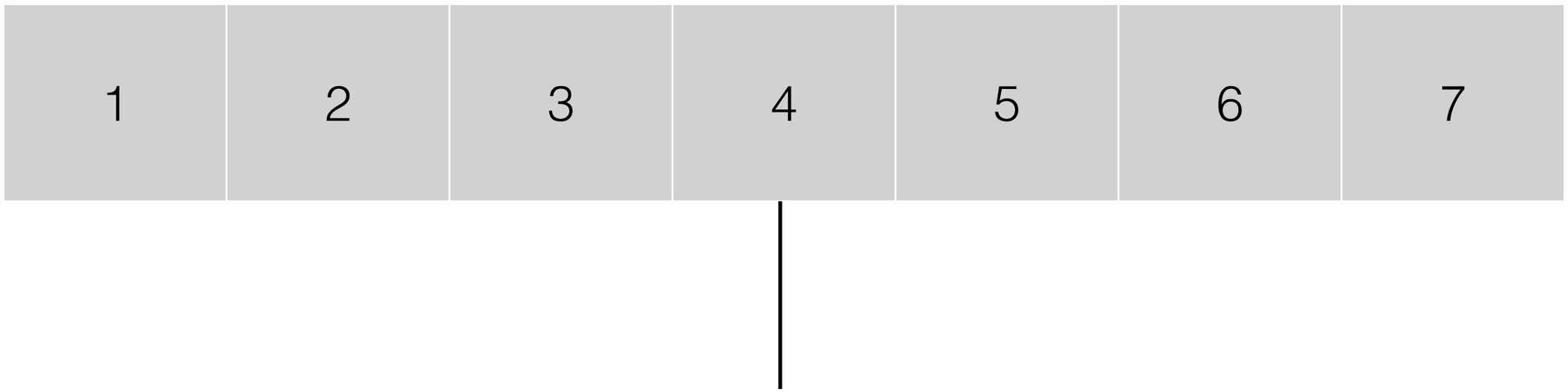
If we guess to high, we try again to the left
If we guess to low, we try again to the right

Binary Search

1	2	3	4	5	6	7

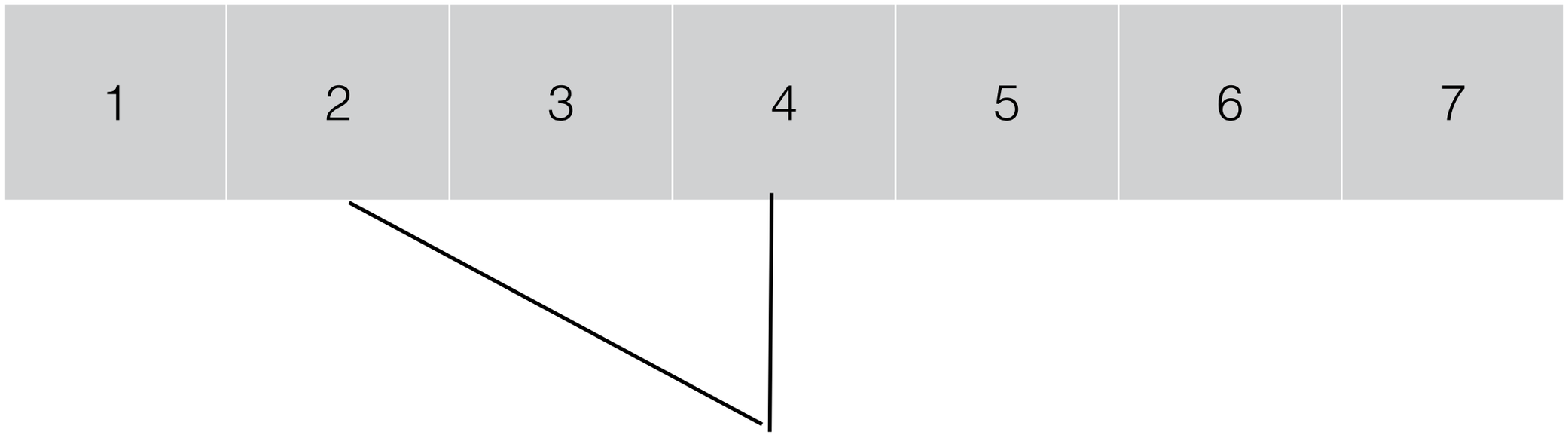
We’re looking for 2

Binary Search



4 is bigger than 2, so go to the left

Binary Search



2 is exactly 2, so we found it!

Complexity of Binary Search

What do you think the Big O of binary search would be?

Complexity of Binary Search

We essentially split the input into two every time

So our Big O has to be O(logn)

Note that O(logn) doesn’t need a base, think about why!

Contact

- My office hours are from 1-3PM in GDC 1.302
- Email: pbae@utexas.edu
- Slides: https://github.com/pybae/slides