Search algorithms

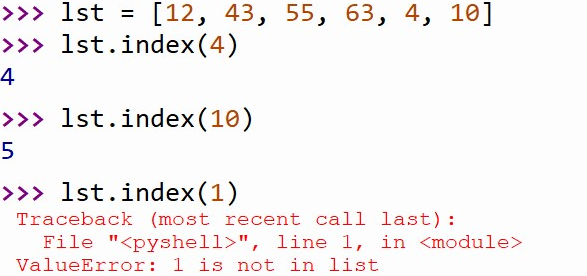
Linear search

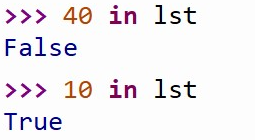
Binary search

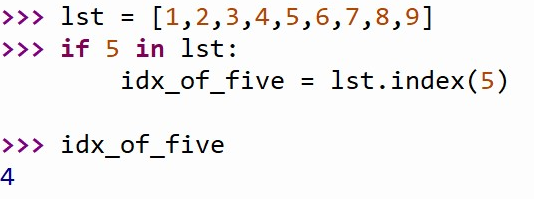
The basic techniques for analysing the efficiency of algorithms

(期末不考)

Linear search

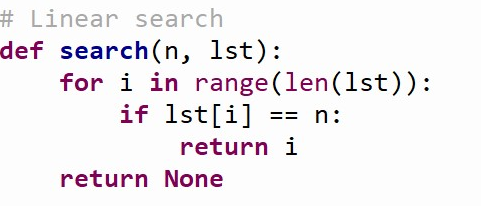






Combine them

The python in and index operations both implement linear searching algorithms



On average, this will save us about half the work

Binary search

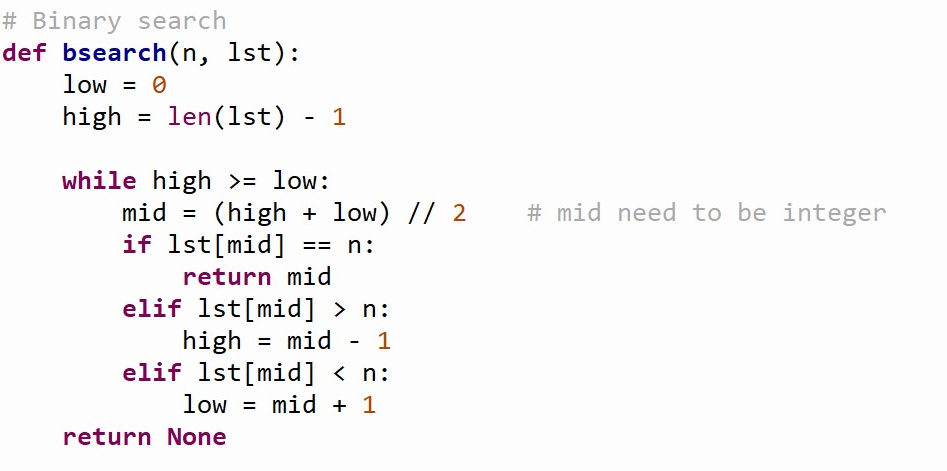
If the collection of data is very large, it makes sense to organize the data somehow so that each data value doesn’t need to be examined. (avoid non-solutions)

Binary search: each time we guess the middle of the remaining numbers to try to narrow down the range.

At each step we are dividing the remaining group of numbers into two parts.

We can use two variables to keep track of the endpoints of the range in the sorted list

Prerequisite: the data is sorted



Linear & binary search

The linear search is easier to understand and implement – to small list

The binary search is more efficient since it doesn’t need to look at each element in the list – for longer list

Comparing algorithms

We can assume that the algorithm with the fewest number of “step” is more efficient.

So how do we count the number of “step” ?

Computer scientists attack these problems by analysing the number of steps (very approximately) that an algorithm will take relative to the size or difficulty of the specific problem instance being solved .

It depends on the comparison statement.

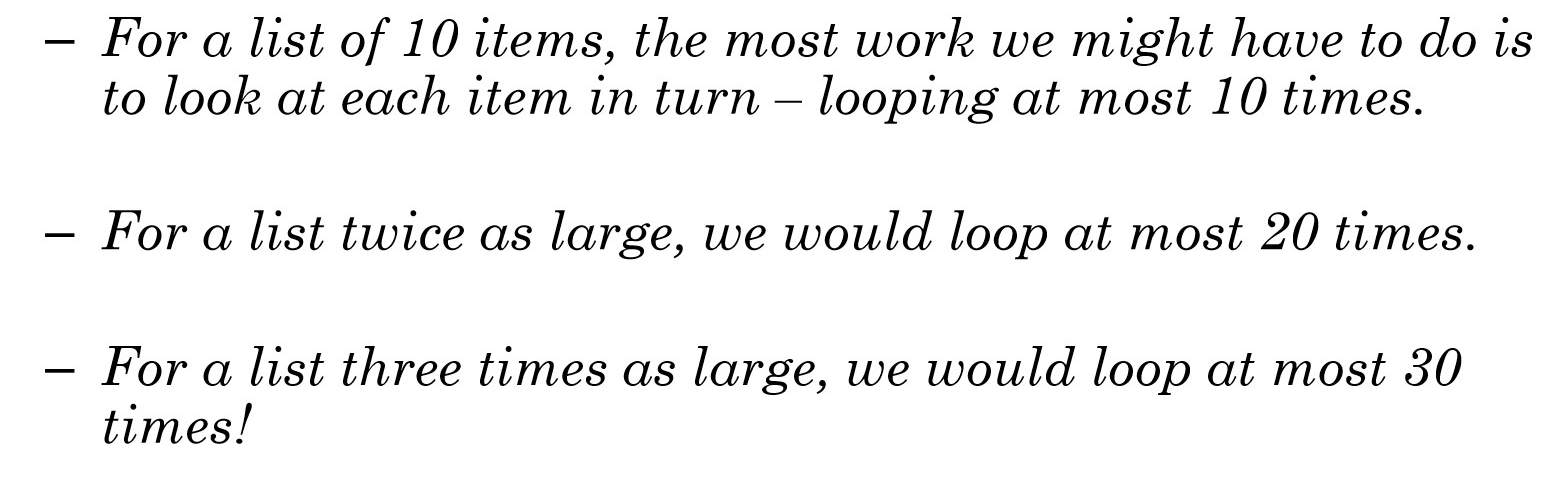
For searching, the difficulty is determined by the size of the collection – it takes more steps to find a number in a collection of a million numbers than it does in a collection of 10 numbers

How many steps are needed to find a value in a list of size n?

In particular, what happens as n gets very large?

Comparing algorithms with linear and binary search

Let’s consider linear search

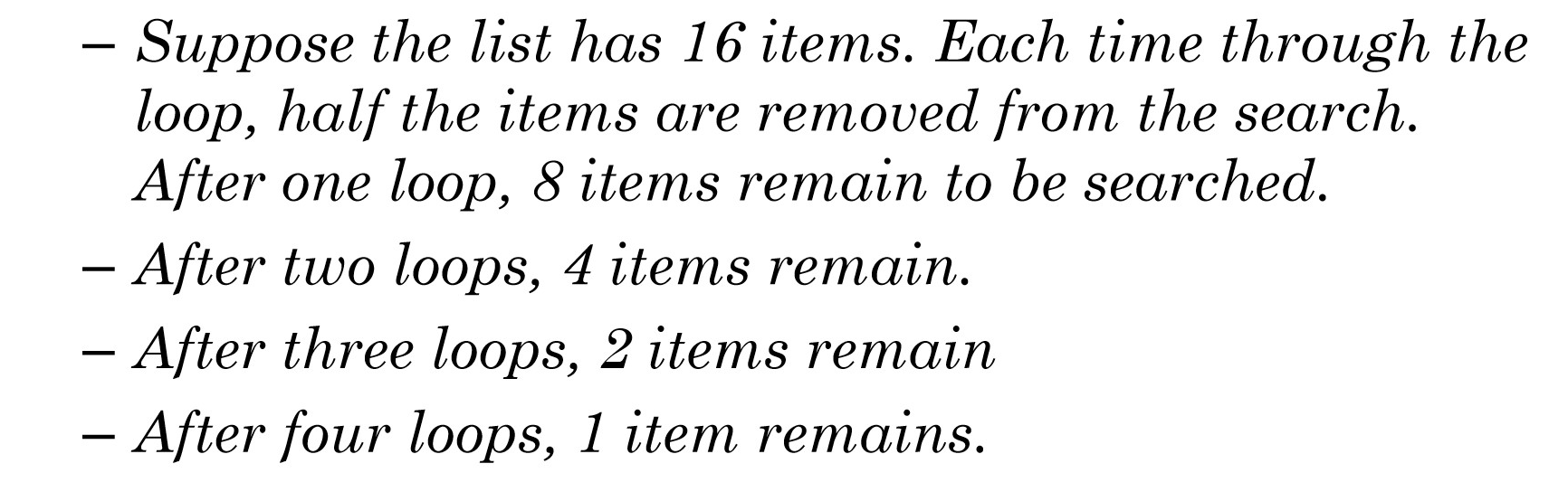


The amount of time required is linearly related to the size of the list, n.

This is a linear time algorithm.

* Notation used is O(N) – order notation

Now let’s consider binary search



If a binary search loops is i times, it can find a single value in a list of size .

* Put another way, if the list has the size N, loops will be required. ->

