**Linear Search - Unordered Array**

- start at first element
- inspect each element in turn
- stop when find X or reach end

If there are N elements to search
- Best case check 1 element
- Worst case check N elements
- If in list on average check N/2 elements
- If not in list check N elements

---

**Linear Search - Ordered Array**

- start at first element
- inspect each element in turn
- stop when find X or find value ¿X¿ or reach end

If there are N elements to search
- Best case check 1 element
- Worst case check N elements
- If in list on average check N/2 elements
- If not in list on average check N/2 elements

---

**Binary Search - Ordered Array**

- start with entire array
- at each step halve the range the element may be in
- stop when find ¡b¿X¡/b¿ or range is empty

If there are N elements to search
- Best case check 1 element
- Worst case check log2(N)+1 elements
- If in list on average check log2(N) elements

How fast does log2(N) Grow?
- log2(10) = 3.3
- log2(1000) = 10
- log2(1000000) = 20
- log2(1000000000) = 30
- log2(1000000000000) = 40

---

**Sorting**

*Sorting* is the process of arranging a list of objects into a specific order. The purpose of sorting is to facilitate the later search for members of the list. For example:

- Words in a dictionary are alphabetically sorted, so that they can be found quickly.
- Books in a library are sorted according to a subject index system. Using this index, a book can be quickly located.

This problem has been extensively studied and many algorithms proposed. If there are N elements to search.

Simple algorithms tend to require a number of operations proportional to N*N
Better algorithms require a number of operations proportional to N*log(N)
If $N$ is large $N \log(N)$ is much smaller than $N^2$

How fast does $N \log 2(N)$ grow?

- $10 \log 2(10) = 33$, $10 \times 10 = 100$
- $1000 \log 2(1000) = 100$, $1000 \times 1000 = 1000000$
- $1000000 \log 2(1000000) = 2000000$, $1000000 \times 1000000 = 1000000000000$

Hence if you have a large number of elements to sort you need a good algorithm.