Figure 20.1 The speed of "linear search".

How long does this take? As always, we need to ask two questions:

1. What operations are we counting?
   —comparisons (there aren't any moves)
2. How many of them are there?

To answer the second question, consider two example lists:

Richard, Mary, Alice, Bob	$\longrightarrow$	1 comparison
Mary, Alice, Bob, Richard	$\longrightarrow$	4 comparisons

For a list of length N,

- the $\boxed{\textit{best}}$ case is $\boxed{\text{1 comparison}}$,
- the $\boxed{\textit{worst}}$ case is $\boxed{N \text{ comparisons}}$, and
- the $\boxed{\textit{average}}$ case is $\boxed{N/2 \text{ comparisons}}$.

Suppose we use this algorithm to search for a name
in the Toronto phone book

- $N \approx 1,000,000$
- ˜0.01 seconds/comparison

The *average* time for a search is

- $N/2 \times 0.01 = 500,000 \times 0.01 = 5000$ seconds
- more than an hour!

This just isn't acceptable. $\boxed{\text{We need a better algorithm.}}$