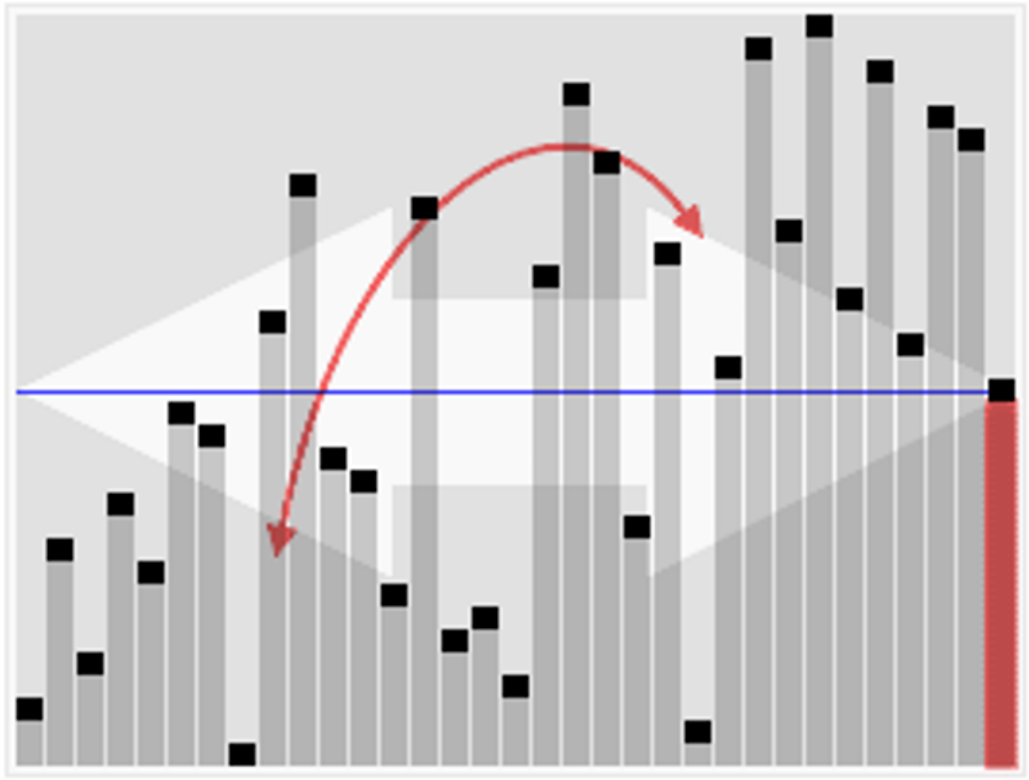


Quicksort

Analiza Algoritmilor

Image: https://en.wikipedia.org/wiki/File:Sorting_quicksort_anim.gif

Overview

```
quicksort(A, left, right):
    if (right - left <= 1)
        return

    pivot_index = choose_pivot(A, left, right)
    partition(A, left, right, pivot_index)
    quicksort(A, left, pivot_index)
    quicksort(A, pivot_index + 1, right)
```

Recurrence relation

Best case scenario (each time we select the *median* as the pivot):

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

Worst-case scenario (each time we select the *min*/*max* as the pivot):

$$T(n) = T(n-1) + \Theta(n)$$

Splitting 25-75

Each time we select a pivot that yields a 25-75 split:

$$T(n) = T\left(\frac{n}{4}\right) + T\left(\frac{3n}{4}\right) + \Theta(n)$$

How many such elements are there?

50%

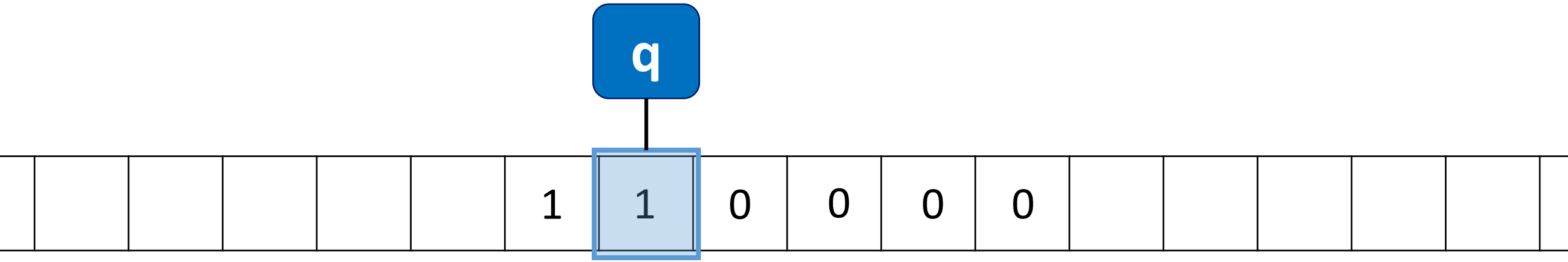
Random algorithms

Coin flips are enough:

- Objects can be represented by binary strings
- A binary strings is just a sequence of 0s (heads) and 1s (tails)

Images: https://romaniancoins.org/ro1leu1870.html

Probabilistic computation



	0	1	□
q	$q, 0, \rightarrow$	$q, 1, \rightarrow$	$p, \square, \leftarrow$
p	$Y, 0, -$	$N, 1, -$	$N, \square, -$

	0	1	□
q	$p, 1, \leftarrow$	$p, 1, \rightarrow$	p, □, ←
p	N, 0, –	$N, 1, -$	Y, □, –