Expected Value

\[ \sum \text{value of random var with that possibility } \times \text{probability} \]

\[ = \sum x \cdot Pr[\text{random var } = x] \]

x that can occur
Counting Sort

$n$ items each integer (or mapped to one) $O_\leq 1, \ldots, k-1$

Do this in array in a stable manner

$O(n+k)$
Radix Sort

\[ n = 7 \]
\[ d = 3 \]
\[ b = 10 \]

assumed \( d \) based \( b \) digits

\( n \) items to sort

From least to most significant digit

apply counting using that digit

sorted

\[ \mathcal{O}(d \ (n+b)) = \mathcal{O}(d \ (n+b)) \]
for (int d = 0; d < numDigits; d++) {
    Arrays.fill(count, 0);
    for (Object x : from)
        count[digitizer.getDigit((E) x, d)]++;
    for (int i = 1; i < b; i++)
        count[i] += count[i-1];
    for (int i = getSize()-1; i >= 0; i--)
        to[count[digitizer.getDigit((E) from[i], d)] = from[i];
    Object[] temp = from; from = to; to = temp;  //swap
```
Sort 1,000,000 social security numbers.

base 10, 9 digits

Counting sort has time complexity $O(n \log n)$

option 1  $xxxxxxxxx 9 10$

option 2  $xxx | xxx | xxx 3 1000$

option counting sort $xxxxxxx 9 10^9$
1,000,000 social security #5, counting sort 3n+2b statements

\[
\frac{d}{b} \quad \frac{\text{# statements}}{9 \quad 10} \quad 9 \left(3 \cdot 1,000,000 + 20\right) = 2,000,0180
\]

\[
3 \quad 10^3 \div 1000 \quad 3 \left(3 \cdot 1,000,000 + 2000\right) = 9,006,000
\]

\[
1 \quad 10^9 \quad 1 \left(3,000,000 + 2 \cdot 10^9\right) = 2,003,000,000
\]

\[
2 \quad 5,4 \quad 6,220,000
\]
$r$ "digits" per radix

$$\frac{d}{r} (c, n + c_2 10^r)$$

# statements

let $10^r = n$

$\max_{d \geq s}$

base $n$
Bucket sort

Non-comparison based sort that works for reals

Place each element into a bucket
Sort each bucket
- N buckets
- place into buckets
- apply insertion sort over entire array
Set ADT

- insert elements
- locate element
- remove element

+ any order iteration, ...
Hold a set of phone numbers organized by area code.

insert phone #

locate all phone #s with a given area code

remake phone # or area code
Direct Addressing

Data structure

(eg. dynamic array)

holds all phone numbers with that area code
Large universe of possible tags/elements

$U \leq 10^9$

Hash function

5000 students

10,000 array

9999